AD-A040 636 HONEYWELL INC MINNEAPOLIS MINN SYSTEMS AND RESEARCH --ETC F/G 17/2
APPLICATION OF ASSOCIATIVE PROCESSING TECHNIQUES TO AN INTEGRAT--ETC(U)
JUN 76 H G SCHMITZ, T L SAXTON, C C HUANG DCA100-75-C-0048

UNCLASSIFIED NL

1 OF 2
ADA
040636

ADA040636

10

NW

APPLICATION OF ASSOCIATIVE PROCESSING TECHNIQUES TO AN INTEGRATED VOICE/DATA SWITCHED NETWORK

FINAL REPORT

APRIL 1975 – JUNE 1976

DDC
JUN 16 1977
C

by

H.G. Schmitz

T.L. Saxton

C.C. Huang

J.A. White

for

THE DEFENSE COMMUNICATIONS AGENCY

DISTRIBUTION STATEMENT A
Approved for public release;
Distribution Unlimited

AD No.
DDC FILE COPY

Honeywell

Systems & Research Center

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER

2. GOVT ACCESSION NO.

3. RECIPIENT'S CATALOG NUMBER

9

4. TITLE (and Subtitle)

6 APPLICATION OF ASSOCIATIVE PROCESSING TECHNIQUES TO AN INTEGRATED VOICE/DATA SWITCHED NETWORK.

5. TYPE OF REPORT & PERIOD COVERED

Final Report.
April 1975 - June 1976

6. PERFORMING ORG. REPORT NUMBER

7. AUTHOR(s)

10 H. G. Schmitz; et. al. T. L. Saxton, C. C. Huang J. A. White

8. CONTRACT OR GRANT NUMBER(s)

15 DCA 100-75-C-0048 new

9. PERFORMING ORGANIZATION NAME AND ADDRESS

Honeywell Systems & Research Center
Mineapolis, Minnesota

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS

Program Element - 33126K

11. CONTROLLING OFFICE NAME AND ADDRESS

Defense Communications Agency
8th Street and South Court House Road
Arlington, VA 22204

12. REPORT DATE

11 June 1976

13. NUMBER OF PAGES

171

14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)

12 184p.

15. SECURITY CLASS. (of this report)

Unclassified

15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

DDC
JUN 10 1977
C

18. SUPPLEMENTARY NOTES

None.

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Communication
Associative Processing
Computers
Switching
Coding
Networks
Integration

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

Recent Defense Communications Agency studies have shown the desirability of an all-digital, switched network which integrates on a single transmission line, voice interactive and bulk data. This document describes the effort and results of a study directed at specifying the network structure and types of nodes required for such a network. Specific emphasis was placed on investigating and evaluating the feasibility of applying associative and parallel processing techniques to all-digital, switched, integrated networks. Results of this contract indicate that, using a line integration technique OVER

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

402 349

Unclassified LB

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

line the one described in this report, such a network definition is possible and that parallel and associative processing techniques can be used in its implementation. Areas requiring further investigation are also identified.

FOREWORD

This study was conducted for the Defense Communications Agency under Contract No. DCA100-75-C-0048 by the Honeywell Systems and Research Center, 2600 Ridgway Road, Minneapolis, Minnesota. The report covers the work performed from April 1975 to June 1976.

The following Honeywell personnel made significant contributions to the study effort reported herein: Dr. H. G. Schmitz, Principal Investigator; Mr. T. L. Saxton, Co-Principal Investigator; Dr. C. C. Huang and Mr. J. A. White, Investigators.

Dr. Cass DeFiore served as contract monitor for the Defense Communications Agency.

ACCESSION for
NTIS White Section
DDC Buff Section
UNANNOUNCED
JUSTIFICATION
BY
DISTRIBUTION/AVAILABILITY CODES
Dist. AVAIL. and/or SPECIAL
A

ABSTRACT

Recent Defense Communications Agency studies have shown the desirability of an all-digital, switched network which integrates on a single transmission line, voice, interactive, and bulk data. This document describes the effort and results of a study directed at specifying the network structure and types of nodes required for such a network. Specific emphasis was placed on investigating and evaluating the feasibility of applying associative and parallel processing techniques to all-digital, switched, integrated networks. Results of this contract indicate that, using a line integration technique like the one described in this report, such a network definition is possible and that parallel and associative processing techniques can be used in its implementation. Areas requiring further investigation are also identified.

CONTENTS

CONTENTS

CONTENTS

CONTENTS

LIST OF FIGURES

LIST OF FIGURES (Continued)

LIST OF FIGURES (Continued)

LIST OF FIGURES (Concluded)

LIST OF TABLES

SECTION 1

INTRODUCTION

Recent DCA studies have shown the desirability of an all-digital, switched network which integrates voice, interactive, and bulk data for the 1980's. At present, numerous distinct communications networks provide these services, independent of one another. Each network serves the distinct set of requirements imposed by a unique user class. Examples of such networks include the AUTOVON and AUTODIN systems maintained by the Defense Communications Agency (DCA). Additionally, it has been established that the primary cost is, and will continue to be, associated with the transmission segment of the network. The objective of this study has been to formulate methods of optimally integrating voice and data traffic in such a way that maximum sharing of network resources is provided to the economic benefit of all user classes.

This study takes advantage of new and unconventional computer architectures to implement control and switching functions. Specifically, the study was aimed at investigating and evaluating the feasibility of applying associative and parallel processing techniques to all-digital, switched, integrated networks.

The effort had two main thrusts towards a common goal. First, a "top-down" system design approach was used. It was based on a functional and performance baseline that insured commonality and modularity in the implementation. Meanwhile, a "bottoms-up" approach was also considered in order to provide evolutionary rather than revolutionary growth and transition from existing

and near-future network plans and equipment to the all-digital, integrated network. The two approaches converged to define an integrated network concept employing associative and parallel processing techniques.

The major objectives of the study program were two-fold:

1) Functionally design an all digital, switched, integrated network and identify the resultant network structure in terms of node types and functions

2) Evaluate nodal architectures incorporating associative and parallel processing techniques optimized for the concept and define basic component building blocks

The study program was conducted as a series of five tasks designed to emphasize key issues, ensuring the best solution to the problems encountered.

Task 1: Establish User Requirements Baseline

Task 2: Functional Design of Integrated Switch

Task 3: Analysis of Integrated Switch Requirements

Task 4: Application of Architectures

Task 5: Specification of Selected Architecture

Section 2 of this report presents the user requirements baseline which was developed during the execution of Task 1. Three user categories were identified and their requirements characterized in this task.

The results of Tasks 2 and 3 are combined and presented in Section 3 of this report. Here, the functional characteristics of an integrated, switched network, like network structure and call control procedures are enumerated. These characteristics were formulated in response to the user requirements established in Task 1. Tasks 2 and 3 constituted the bulk of the effort conducted on this study.

Section 4 documents the results of the "Application of Architectures" and "Specification of Selected Architecture" tasks by presenting the node data path structure and an associative switching technique for handling line switched calls. Also included in this section is a description of the hardware structure necessary for processing packet-switched calls and data.

Section 5 gives a summary of the work performed to date and recommendations for future work.

SECTION 2

USER REQUIREMENTS BASELINE

This section presents baseline requirements as two categories:

- Network requirements
- User requirements

Within each category, performance requirements are also presented. When possible, specific reference is made to existing operating procedures and protocols.

The requirements were developed by consolidating information from three sources: 1) documentation of existing or planned systems, 2) literature on new network designs, and 3) DCA planning information. The resulting user requirements baseline was generated in a form suitable for definition of an integrated network.

NETWORK CHARACTERISTICS

Application of Associative Processing

The network design approach should take maximum advantage of associative processing techniques, which are meant to encompass parallel processing and associative memories, in the switch design as well as for multiplexing and compression.

Network Structure

A hierarchical network is envisioned to provide for a tandem level trunk switched network interfacing to users via local access.

PRECEDING PAGE BLANK-NOT FILMED

The network switches will use common equipment whenever possible for the data path connections and signalling, supervision, and control paths as well as for switch control. A hybrid circuit/packet switch is envisioned.

Network Users

Network users fall into the general categories of voice (clear and secure), narrative/record message, interactive data, and bulk data. Other specific uses will include:

- Providing the backbone network for AUTODIN I hosts
- Providing an alternate network for AUTOVON subscribers, primarily for data transmissions
- Providing ports for terminal access to ARPANET-like network services
- Interconnecting various data terminals and computer services

Interconnection of Incompatible Services

Various subscribers and services with incompatible data rates and protocols will be afforded interconnection whenever feasible through buffering and translation services provided by the network.

Transparency

The network should be transparent to high level protocols already established, tested, proven, and operable, as currently implemented on the ARPANET [1]. HOST-HOST protocols and higher are specifically external to the network and thus must be independent of the network design and structure.

USER LEVEL REQUIREMENTS

User Classes

Three distinct classes of users have been defined.

Class I: On Demand, Fixed Delay -- This class of traffic is characterized by calls comprised of continuous transmission (perhaps one long transaction) which the network accepts without delay and forwards with a fixed delay (voice, facsimile, video).

Class I consists of two subclasses:

> Class IA: Noncompressible
> Calls in this subclass require continuous transmission and bit count integrity.
>
> Class IB: Compressible
> Calls in this subclass may be compressed to reduce their bandwidth requirement. Pause removal is one possible compression technique. Bit count integrity is not required.

Class II: On Demand, Variable Delay -- This class of traffic is characterized by calls comprised of discontinuous bursts of short transactions which can tolerate variable delays so long as the average delay (or maximum worst case delay, depending on the design criteria) is near real time and does not exceed an acceptable value, T_D. Two subclasses, A and B, are further defined. Class A includes interactive data and has a relatively short delay, T_{DA}. Class B is for narrative/record message traffic with a longer delay, T_{DB}.

Class III: As Available, Variable Delay -- This class of traffic is characterized by calls comprised of long non-real time transactions which may be interrupted and may be delivered with significant variable delays (bulk data).

Types of Service

The types of service provided to users depend upon the user's precedence level. The precedence level is based on priority and class. Various schemes for assigning precedence are possible, depending on the rules established for allocating bandwidth. Specifically, a rule which will allow allocation of the entire bandwidth to the highest precedence user will lead to significantly different schemes than one which guarantees a minimum fraction of the total bandwidth for each level and for each user class. The amount of bandwidth allocated to each class will be parameterized so that it may be tailored to fit each switch installation's traffic requirements. This will allow a potential reservation for each class ranging from none to any percentage. In the case of no reserved bandwidth for a class, that class would still be guaranteed a minimum service level via preemption.

Five levels of precedence may be used to provide the following types of service for Class IA:

Class IA -- Priority	1	Non-blocking (always accepted)
	2	Accepted without delay or blocked; probability of blocking = P_{A1}
	3	Accepted without delay or blocked; probability of blocking = P_{A2}
	4	Accepted without delay or blocked; probability of blocking = P_{A3}

Class IA -- Priority 5 Accepted without delay or blocked; probability of blocking = P_{A4}

where $P_{A1} < P_{A2} < P_{A3} < P_{A4}$.

Classes IB, IIA, IIB, and III may be combined into a single precedence structure as shown in Table 1.

Table 1. Precedence Structure

PRECEDENCE LEVEL	PRIORITY	DESCRIPTION
1	0	Control
2	1	Class IB
3	2	Class IB
4	3	Class IB
5	4	Class IB
6	5	Class IB
7	1	Class IIA
8	1	Class IIB
9	1	Class III
10	2	Class IIA
11	2	Class IIB
12	2	Class III
13	3	Class IIA
14	3	Class IIB
15	3	Class III
16	4	Class IIA
17	4	Class IIB
18	4	Class III
19	5	Class IIA
20	5	Class IIB
21	5	Class III

The corresponding types of service are:

Class IB -- Priority	1	Non-blocking (always accepted)
	2	Accepted without delay or blocked; probability of blocking = P_{B1}
	3	Accepted without delay or blocked; probability of blocking = P_{B2}
	4	Accepted without delay or blocked; probability of blocking = P_{B3}
	5	Accepted without delay or blocked; probability of blocking = P_{B4}

where $P_{B1} < P_{B2} < P_{B3} < P_{B4}$.

Class IIA -- Priority	1	Accepted with delay = T_{DA1}
	2	Accepted with delay = T_{DA2}
	3	Accepted with delay = T_{DA3}
	4	Accepted with delay = T_{DA4}
	5	Accepted with delay = T_{DA5}

where $T_{DA1} < T_{DA2} < T_{DA3} < T_{DA4} < T_{DA5}$.

Class IIB -- Priority	1	Accepted with delay = T_{DB1}
	2	Accepted with delay = T_{DB2}
	3	Accepted with delay = T_{DB3}
	4	Accepted with delay = T_{DB4}
	5	Accepted with delay = T_{DB5}

where $T_{DB1} < T_{DB2} < T_{DB3} < T_{DB4} < T_{DB5}$.

Class III -- Priority	1	Accepted but interruptable, delay = T_{D1}

Class III -- Priority	2	Accepted but interruptable, delay = T_{D2}
	3	Accepted but interruptable, delay = T_{D3}
	4	Accepted but interruptable, delay = T_{D4}
	5	Accepted but interruptable, delay = T_{D5}

where $T_{D1} < T_{D2} < T_{D3} < T_{D4} < T_{D5}$.

For all classes the initial call dialogue is permitted to take place in order to determine the Precedence Level (PL) of the user. For Classes II and III, data is not actually preempted. Rather, priority to buffer space is given to higher precedence traffic.

For Class I users, the status of the called subscriber will be determined (unless the total bandwidth available to Class I users of equal or higher precedence is already allocated) and if not busy, the connection will be made. If the total bandwidth available to the requesting Class I user is allocated, the call will be blocked and lost (i.e., the initial call dialogue information will be discarded).

For Class II, the network will provide service equivalent to that provided by the current ARPANET. That is, the network will be transparent to the HOST-HOST protocol. The initial call dialogue will consist of a message from a particular host subscriber to the node. The node, after acceptance of the message, will store the call control information contained therein and return an abbreviated address for future use along with a Request For Next Message (RFNM) when ready. Thereafter, the subscriber may input a message

into the node using the abbreviated address whenever it has received an RFNM from the node.

The call may be cleared by a HOST-NODE message to take down the call or by a NODE-HOST message based on a time-out or critical system component outage.

This will result in destruction of the stored routing data. Note that the purpose of the RFNM is to control network congestion. The RFNM is completely separate from and transparent to the HOST-HOST protocol.

Class III users are offered service on the same basis as Class II except that, since messages are longer, the network may arbitrarily interrupt input in the middle of a message by, for example, stopping a clock provided to the user for timing his input.

User Class Characteristics

Users in a given class may establish calls only with other users in the same class, i.e., no inter-class communication is possible.

Class I -- A number of subclasses can be defined based primarily on the data rate required for I/O and the technique used for compression (if any).

Subclass

User Type	Clear	Secure	Data Rate
Voice	VoC	VoS	2.4K,8K,16K,64K
Facsimile	FC	FS	4.8 to 300K
Video	ViC	ViS	$\geq$ 150K

Clear signals may be compressible (Class IB). Secure signals are characterized by the lack of detectable signal presence and so are not compressible. For voice, the 8K and 16K bps data rates are assumed to be generated by a CVSD processor and the 64K bps by an 8-bit PCM encoder taking 8000 samples per second. The clear CVSD voice is further compressible through recognition and removal of pauses. The clear PCM is further compressible through pause removal as well as redundancy removal within talk-bursts. Video and facsimile are treated as noncompressible (Class 1A).

Additional Class I characteristics include:

- Call lengths are exponentially distributed with a 5-minute mean.
- Cross network connection time is 3 seconds nominal after completion of call initiation dialogue.
- Maximum cross network delivery after connection is 250 milliseconds.
- There is no error control.
- Blocked calls are lost.
- Bit count integrity of secure signals is important.
- All received data streams from access lines at the node are synchronized to the local switch clock. In the case of digital local subscriber loops, the transmit signal will provide clocking to the subscriber equipment through, for example, clock recovery from a bipolar coded signal.
- All users are permanently assigned network ports.
- The initial call dialogue is generically the same for all subclasses. Subscriber addressing will be accomplished

via a numbering scheme equivalent to AUTOVON with the capability to specify preemption level and security classification as additional dialed digits. The identity of the calling subscriber and his subclass as well as data rate is assumed to be known by the local switch from information stored with the port assignment. The following two types of signalling for the call initiation dialogue will be accommodated:

1. (R1) in band SF to E lead, M lead output, and
2. (CCITT4) in band DTMF to E lead, M lead output.

Signalling type is fixed for each port assignment.

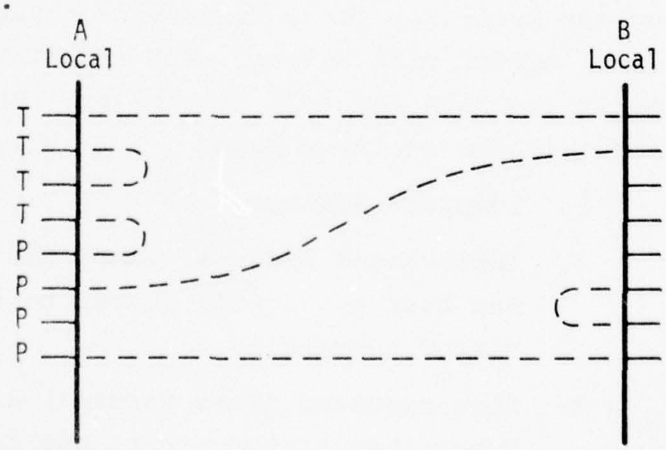
Class IIA -- The interactive data class includes:

- Terminal-to-terminal, where one user is talking to another in real time as in computer mediated conferencing.
- Terminal-to-computer, as in inquiry/response and time-sharing systems.
- Computer-to-computer, where a reasonably tight time response criteria exists, as in an on-line resource sharing network.

Interactive data is characterized by short messages. In both cases a logical connection is required prior to information transfer. The requirements for this class are listed below:

- Cross network connection time is 30 seconds nominally, with a nominal one-second cross network delivery time.
- Expected message size is assumed to range from as few as three 8-bit characters plus header up to a maximum of 40,000 characters, with length exponentially distributed and a mean of 2000 bits.

- A virtual interconnect may last several hours. However, operation is typically half-duplex so that only simplex switching is required at a node; i.e., each direction must be switched independently.
- Variable input rate is from 75 bits per second to 100K bits per second.
- Error control is required for each packet in a link-by-link fashion (i.e., not end-to-end) and is required on access lines for all but Mode IIA.
- Bit count is important.
- Since data exchanges between terminals and computers will be effected within this class by the network, a single call may require one line interface discipline and protocol at the terminal and quite a different one at the computer where some service residing within the computer is being sought by the terminal user. The network must provide the means to accomplish this. In addition, various terminal types operating at different data rates must be flexibly accommodated. The types of interconnections required are shown below, where T stands for Terminal and P stands for Program[1].



Terminal users may access the network via either a dial-up port (as on ARPANET TIPS) or fixed port assignment. Following establishment of a connection via dial-up through a commercial network or some other equivalent network external to the local switch, the terminal type and data rate will be established by the user typing a letter unique to that terminal. From this the line interface discipline, protocol, and data rate peculiar to that terminal will be established. From this point on, the call dialogue will be the same as for a terminal with a fixed port assignment whose terminal type, protocol, and data rate are already known.

The allowable line interface disciplines shall conform to Mode IIA as described in Reference 2. Data rates from 75 bps to 1200 bps are permitted. Identifiable, distinct terminal devices (e.g., printer, cassette tape, etc.) may be addressed. Terminals in turn may be addressed as devices associated with a host, even though the host may be only a virtual host as described elsewhere.

The initial call dialogue may take on several forms.

- After input of the unique letter for a dial-up terminal or the break key for a fixed-port assigned terminal, the local switch will respond with a "HELLO" message to request the data for call initiation. The address may be input in one of three ways:

    1. Complete address
    2. Abbreviated address, where the complete address has been previously stored to facilitate abbreviated addressing
    3. None required (This terminal always has single subscriber transactions, and the complete address is prestored.)

- The computer will, in general, appear as multiple logical channels on a single access line interface with a dedicated port assignment equivalent to that used for the ARPANET HOST. Each message input or output must contain sufficient information to identify its logical channel number. The interface line discipline will adhere to the Mode VI standards as set forth in Reference 3. The ARPANET HOST to IMP protocol is assumed for this interface.

Class IIB -- Narrative/record messages are similar to interactive data except in the following ways:

- Longer connection and delivery times are acceptable (i.e., 5-minute mean delivery time for Priority 1 up to 16 minutes for Priority 5).
- Average message length is 20K bits.
- The line interface discipline is Mode I as specified in Reference 4. All protocols and formats must conform to those specified in Reference 2 for AUTODIN I (i.e., the Type B - Packed Binary Segment Leader referred to in Appendix B of that reference). However, a method for flow control is required to limit message input to the network. In addition, a call initiation dialogue is required.

Class III -- Bulk data implies looser time constraints (i.e., a real time response is not required) and long messages. Such data is generated by:

- Remote batch systems
- Computer-to-computer transmissions, such as periodic file maintenance or update.

These are the requirements for this class:

- Cross network connection time is 30 to 60 seconds with non-critical delivery times (i.e., from 5 minutes for Bulk (1) to 4 hours for Bulk (2)).
- Message statistics are similar to interactive data except that the maximum message size is 10^8 bits with a much higher mean length.
- Variable input rate from 4.8K bits per second to 1M bits per second is desired.
- Error control is required. A 32-bit CRC per packet shall be provided as a minimum.
- Bit count integrity is important.
- Mode VI line interface discipline as specified in Reference 3 is required.
- Addressing and call initiation is similar to that described for Class IIA.

Line Interface Discipline

Mode IIA -- Character asynchronous

- Delete (add) start-stop bits on input (output)
- Seven-bit ASCII and parity
- No code conversion required
- No error recovery or line discipline provided (It is the end user responsibility.)
- Data rates of 150, 300, and 1200 bps

Mode I - Character synchronous

- Seven-bit and parity ASCII
- Character parity and block parity checking along with retransmission of errored blocks provided
- Line discipline as stated in Chapters 1 through 5 of the DCS AUTODIN Interface and Control Criteria, DCAC 370-D175-1, Continuous Mode [4]
- Control characters recognized and responded to
- Idle SYN character deletion (insertion) on input (output)
- Bit-oriented subscribers must convert to seven-bit plus parity ASCII to address Mode I subscribers -- i.e., no code conversion
- Classmark analysis (precedence level and signalling type)

Mode VI -- Binary synchronous

- Line discipline defined in ANSI ADCCP -- Independent Numbering ANS X3S34-589 [3]
- Data rates from 4800 bps to 1M bps
- A 32-bit cyclic redundancy check (CRC) provided
- Classmark analysis (precedence and signalling type)

Line Formats

This subsection describes the formatting of the information entered into and received from the network by a subscriber. Information as used here includes the instructions to the network for handling and routing as well as the data to be exchanged between subscribers.

Class I -- Class I users have very straightforward interfaces. All routing and handling information is specified by rather restrictive signalling means. The node will convert the signalling information to a segment leader format, where it will be handled as any other input segment. Since addressing will be done with an AUTOVON compatible numbering scheme, the node will have to convert to a network directory entry, as used for all segment routing, with reconversion back to AUTOVON-compatible addresses at the destination node. Take-down is accomplished by signalling from either the calling or called subscriber.

Class II and III -- Formats for these classes will conform to those specified in Reference 2 for a packet switched network. In order to avoid inventing new terminology and/or confusing meanings of existing terms, the basic element of exchange between a subscriber and the network is the segment. The segment is composed of a leader followed by text. Error control between subscriber and node is required for all but Mode IIA. Variable size segments are permitted, up to a maximum size. Messages between subscribers which exceed the segment size must be broken into segments by the subscriber prior to network entry. The segment leader includes the destination address, security level, precedence level, transmission control code, a segment identifier or logical channel identifier, and a sequence number. Although the content of each acceptable segment leader is the same, the details of the leader depend upon the line format and mode of line discipline.

The five acceptable line formats are:

1. Binary
2. Packed Binary
3. Character
4. Canned Character
5. Character Unclassified

This range of formats will permit efficient operation for complex computer interfaces with binary-oriented data and logical channel multiplexing as well as low-speed, character-oriented, asynchronous, half-duplex operation. When these five line formats are combined with the three line modes, I, IIA, and VI, 11 types of interface formats result. These types are designated type "A" through "K" and are fully described in Appendix B of Reference 2. Detailed explanations of the fields are contained in Section 3.3.3.8.3.1 of Reference 2. Additional addressing functions required to support subscribers are described in Section 3.3.3.8.4 of the same reference.

Two observations can be made at this point:

- Only Types A and B, Binary and Packed Binary, allow multiplexed logical channel operation;
- The type of error control differs according to mode .

Terminal Entered Command Codes -- Types I, J, and K (all Mode IIA) are used for asynchronous terminal access. For these types, certain additional functions are required which can be specified in the CC field of the leader. Since not all terminals have a full ASCII character set, and since more flexibility must be provided to terminal users, Mode IIA types need the capabilities to 1) specify an end of segment (i.e., tell the node when to release a packet to the network), 2) specify echo or no echo from the node, and 3) specify normal connection or interactive session (where an interactive session implies the node will buffer received packets if the subscriber is currently inputting a segment until completion of input, and normal connection implies unmediated output upon receipt). An ASCII alphabetic character will specify the command code as shown in Table 2.

Table 2. Command Code

	Packet Release on Receipt at Source Node of		
	CR	Special Character	N Character
Normal connection without echo	A	E	I
Normal connection with echo	B	F	J
Interactive without echo	C	G	K
Interactive with echo	D	H	L

Call Types -- There are two basic call types permitted. The first is "connection" oriented, where a subscriber sends to or receives from only a single source subscriber which it has previously a addressed or from which it has accepted a connection. An example is a terminal-to-terminal transaction or a terminal-to-computer dialogue, as in a time-sharing system. In the former, both are connection oriented, but in the latter, the computer is not connection oriented. Rather, the computer is using the other basic call type which is message oriented. That is, it accepts traffic from any source on a segment-by-segment basis. This does not imply the computer has no control over segment traffic. Via HOST-HOST protocol, message traffic is regulated.

For connection oriented calls, the sender attaches a leader to the segment in only the initial transaction segment which is used to set up the call. A receiver operating in this mode will accept the first segment only with a leader, after which it will indicate to its node that it can receive no segments other than those from

the subscriber whose call it just accepted. Thus, segment or message receipt is restricted to only one subscriber at a time. The call will remain up until either the calling or called subscriber initiates a call termination dialogue with its node.

For message oriented calls, a leader must accompany every segment although this does not preclude the use of abbreviated addressing in the subscriber-node protocol.

Note that, although a subscriber may be operating in a connection mode for sending, it can be receiving in a message mode, as with a timesharing terminal user who is able to accept mail or messages from any other user on the system although his dialogue on input is exclusively with the computer. Also, as in this same example, a subscriber may change from restricted to nonrestricted receipt of messages by notifying the node to change his call type for receipt.

Additional Service Features

Conference calls for Class IVoC subscribers are desirable. However, the technique by which CVSD processed voice is conferenced is as yet undefined and hence will not be dealt with specifically.

Abbreviated addressing will be provided for all classes.

Roving subscribers should have the ability to modify their homed local switch location.

Local Access Configuration

The node shall provide for the following quantities of user accesses. These figures represent maximum numbers and may be significantly lower, especially at early installations.

Class	Quantity
IVo (C or S)	20-64
IF (C or S)	8
IVi (C or S)	8
IIA -- Terminals	64
IIA -- Computers	8
IIB	1
III	8

Traffic

Class IVo -- All voice traffic is characterized by a Poisson arrival process for call originations, with service provided on a blocked calls lost basis. Holding times are exponentially distributed with a five-minute mean. A Grade of Service (GOS) of 0.05 is assumed. Voice traffic is assumed to originate from multiple PBX's with a maximum of 64 PBX trunks connected.

To provide a GOS of 0.05 overall requires that the sum of probabilities of blocking casued by all potential sources does not exceed 0.05. Potential sources along with their assumed budget allocation (assuming a non-blocking switch) are:

1. PBX trunks 0.025
2. Switch 0.025

For 1, the offered load to the PBX trunks must not exceed 50 erlangs for 64 trunks. For the average holding time of five minutes, this yields 433 served calls and 455 call attempts per busy hour to the PBX. The average arrival rate at the local node is one call every 7.9 seconds. This is the traffic resulting from the assumption that all PBX trunks connect to a single PBX. It would be lower for the same GOS if the trunks were divided between multiple PBX's. Since none of this traffic is to be connected to local subscribers, all of it is assumed to be offered to the exiting trunk network.*

Therefore, the maximum number of trunks that could be utilized is 64, and if this number is provided, trunk availability will not be a source of blocking. Assuming 64 trunks, then, the availability of a cross-switch path connection will be the other source of blocking to which a budget of P (0.025) is allocated.

Class IF -- Fax will present messages to the network with an average length of 4×10^4 bits. Call initiation is assumed to be the same as for digital voice. Error control is not required. At a 4.8K bps rate, a message will last 8.3 seconds. Not more than eight calls will be handled simultaneously. Average call arrival rate is one call every two seconds. The data rate may range from 4.8K to 300K bps.

Class IVi -- Video will present data to the network at a data rate greater than or equal to 150K bps. All other characteristics are assumed similar to Class IVo (Voice).

---

* Even if some traffic were intra-node traffic, based on CONUS AUTOVON statistics, it would represent no more than 10 to 20 percent of the total traffic.

Class IIA -- The traffic generated by each subscriber (i.e., data terminal or computer) in this class is given in Table 3 which was obtained from Reference 5. The input data rate is variable from 75 bps to 100K bps.

Class IIB -- Narrative/record message traffic is assumed to originate only from AUTODIN I switches of which there is not more than one at a local node. Data rates are the same as for Class IIA.

Class III -- Bulk data is characterized by very low arrival rates from a given subscriber or computer. Table 4 presents typical figures taken from Reference 5.

Table 3. Traffic Generated per Terminal During Busy Hour

User Mode	Terminal Data Rates	One-Way Transaction Time (Nominal)	No. of One-Way Transaction per Busy Hour (Nominal)	Average Information Bits Generated (Term)	Message Size (Nominal)	Mode
Data Terminals	Low Speed 150-600 b/s (450 b/s nominal), (Transmit/Receive)-Approx. 2/3 of Terminals	Query -- 4 sec.	20	36 kb	1.8 kb	IIA
		Response -- 40 sec.	20	360 kb	18.0 kb	
	High Speed 2.4-4.8 kb/s (3.6 kb/s nominal), Approx. 1/3 of Terminals	15 sec.	13	702 kb	54.0 kb	VI
Computer to Computer						
Query	Single common	3 sec.	130	187.2 kb	1.44 kb	VI
Response	4.3 kb/s Access line	4 sec.	175	3.36 Mb	19.2 kb	VI

Table 4. Traffic Generated per Terminal During Busy Hour

User Mode	Terminal Data Rates	One-Way Transaction Time (Nominal)	No. of One-Way Transaction per Busy Hour (Nominal)	Average Information Bits Generated (Terminal)	Message Size (Nominal)	Mode
Bulk (1)	4.8 K to 1 M BPS	60 sec.	15	4.32 Mb	288 Kb	VI
Bulk (2)	4.8 K to 1 M BPS	1 hour	2 (per day)	34.5 Mb (per day)	17.28 Mb	VI

SECTION 3

FUNCTIONAL DESIGN AND ANALYSIS OF REQUIREMENTS

This section presents the functional design of an integrated network which is responsive to the requirements defined in Section 2. The design includes the following: operational characteristics, selected technology candidates, and additional functional requirements. Also included is the analysis of all requirements in preparation for the architectural evaluation presented in Section 4.

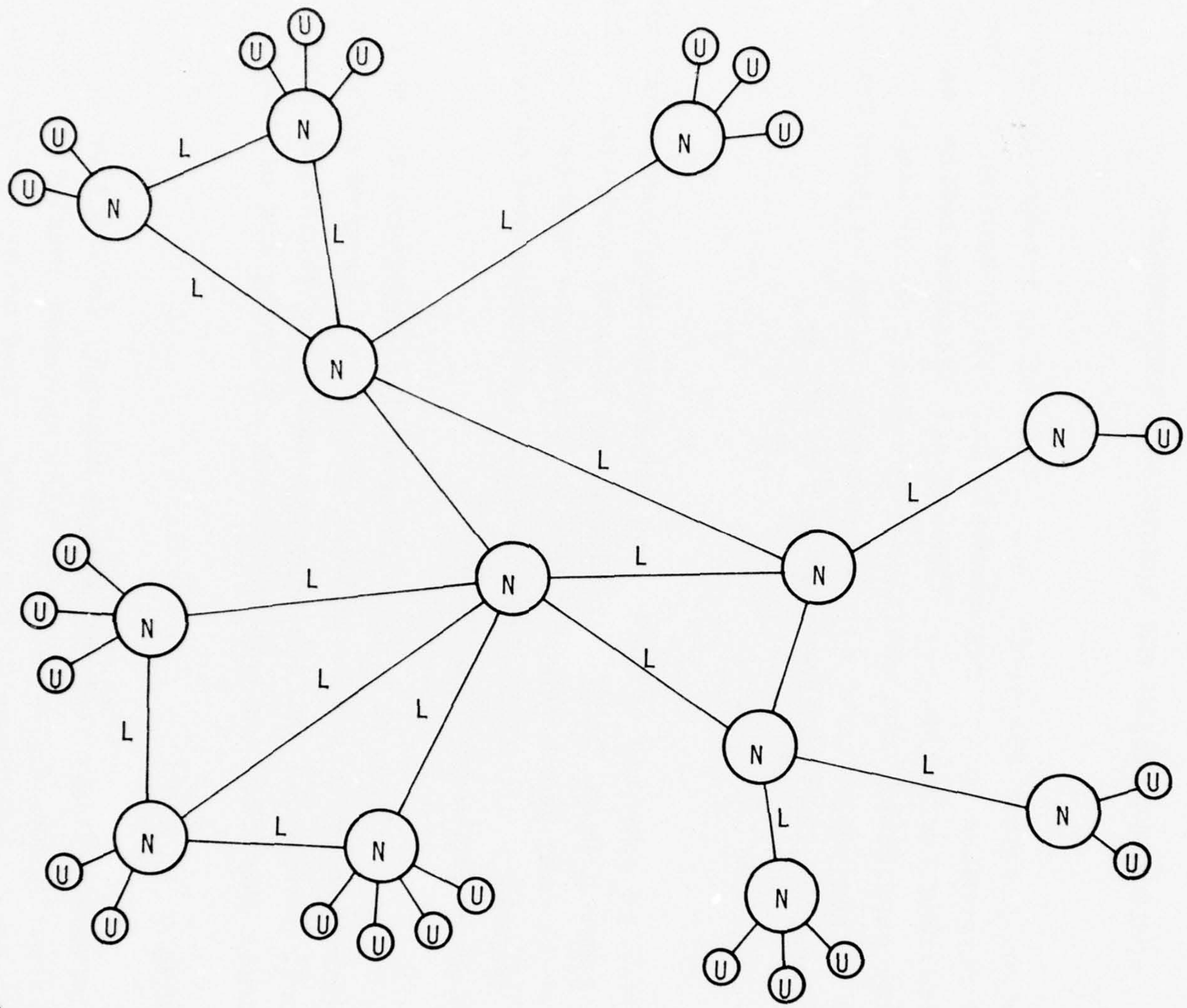
NETWORK

The network is composed of nodes and interconnecting communications links. Nodes provide both user access and network switching. Links may exist in many forms ranging from leased common carrier lines to private microwave satellite relay links. A hypothetical network configuration is depicted in Figure 1.

A block representation of a node is presented in Figure 2. Note that information received on network trunks is separated according to switching required and two separate switching facilities, one for circuit switching and one for packet switching are provided.

FRAME FORMAT

In the proposed integrated voice/data network, information will be transmitted in one of two ways: as line switched data or as packet switched data. All Class IA (on demand, fixed delay, incompressible) traffic will be treated as line switched data where a fixed-path connection is established between the subscribers and remains fixed



N: Node
U: User Access
L: Link

Figure 1. Network Structure

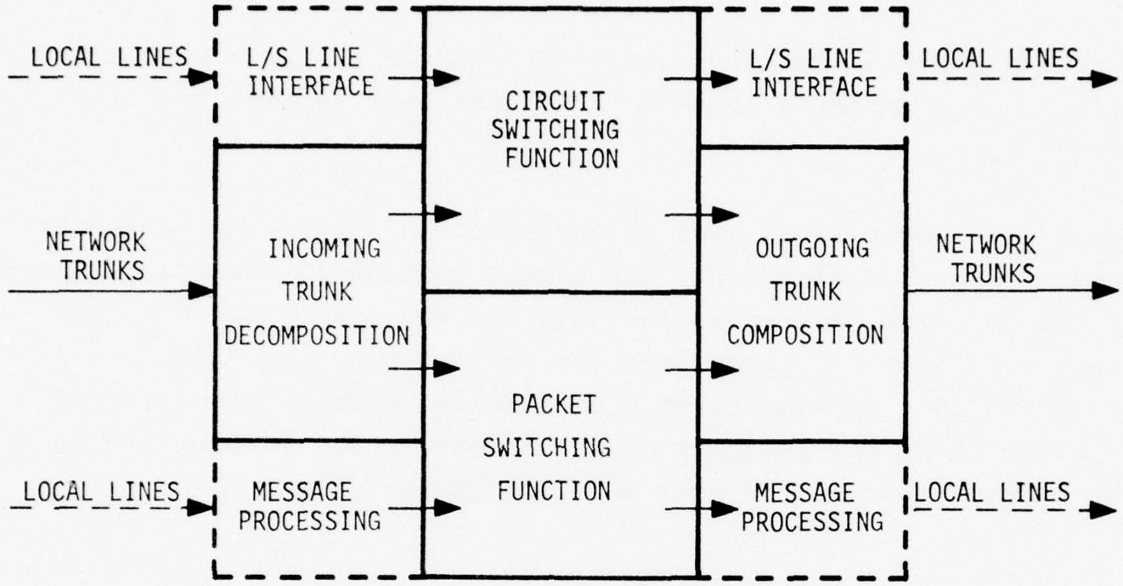


Figure 2. Node Structure

for the duration of the call. All Class IB (on demand, fixed delay, compressed) traffic will be treated in a somewhat similar manner. A fixed-path connection will be established for the duration of the call. However, samples will be packetized at the source node for transmission via a fixed network route. All Class II (on demand, variable delay) and Class III (as available, variable delay) data will be packetized at the source node before being transmitted over the network. The route is not fixed.

The information transmitted over each link of the network is in a form of a binary bit stream. In order to maintain fixed delays for Class I calls, the bit stream over a link is grouped into "frames," where each frame has a fixed number of bits corresponding to the

number of bits transmitted over that link in a fixed, predetermined time. For example, a frame on a T1 link over 10 msec contains 15,440 bits.

The frame format to be used in the proposed network has a general form as shown in Figure 3. The number of bits in the frame depends on the speed of the particular link and the frame length. (The frame time is fixed throughout the network once determined.) A frame consists of a Start-of-Frame marker, a Transition-Synchronization Field and a number of variable-size slots which are used for transmission of L/S call data and packet data.

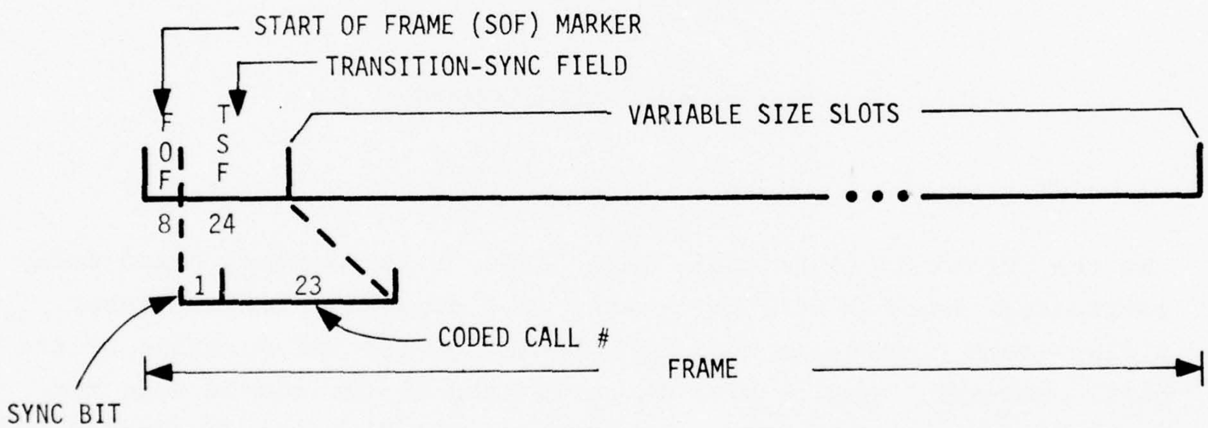


Figure 3. Frame Format

The Start-of-Frame marker (8 bits) is used for frame synchronization acquisition and maintenance. The Transition-Synchronization field contains 1 bit to indicate a transition from L/S to P/S (or vice versa) for some slots in the next frame. Note that the transition

is indicated in the frame prior to its occurrence. The remaining 23 bits contain a 14-bit call number which is coded into the 23-bit form for error protection. The call number identifies the call and, consequently, the transitioning slots.

INTEGRATION

Integration of voice and data on a common transmission facility is accomplished by transmitting information in three ways:

- Incompressible voice samples are transmitted in fixed, variable-size time slots, and
- Compressible voice samples are packetized and transmitted using packet procedures and techniques.
- Data is transmitted as packets which are used to "fill" the remaining available trunk bandwidth (Figure 4).

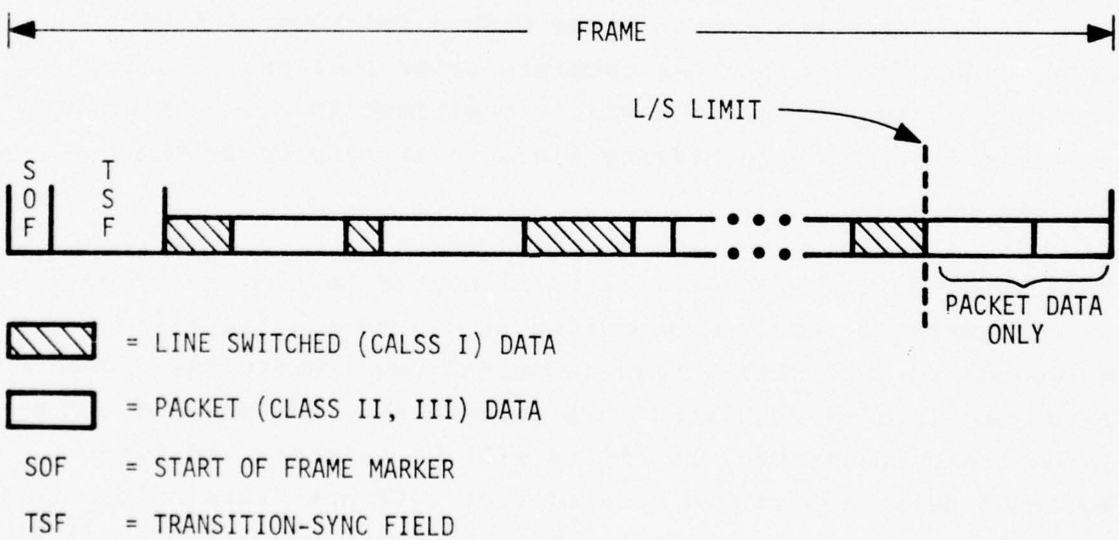


Figure 4. Line Switched/Packet Data Integration

Non-clear (incompressible) voice data (Class IA) cannot be transmitted via packet. If it were transmitted as regular packets with CRC field and allowing retransmission, the retransmission of packets would increase the cross network delay and a large buffer would be required at the destination node to remove network-induced skew. If it were transmitted as voice packets with no CRC field and no retransmission, errors in the voice packet header (24 bits) would result in the loss of bit count integrity.

L/S Data

After an L/S call is set up, variable-size slots (not necessarily contiguous) are allocated for it and remain unchanged for the duration of that call. An integer number of variable-size slots will be used for an L/S call, and a "best-fit" algorithm will be used to minimize the number of slots required per call. Each variable-size slot is an integer number of bytes in length. If an L/S call rate requires n bytes plus a fraction of another byte in the frame, then n+1 complete bytes will be allocated for it. The content of the unused bit positions in the last byte of the last slot can be arbitrary since it is ignored by the receiving node.

Time slots within a frame will be allocated to line switched data on the basis of required bandwidth (i.e., more bytes will be allocated to data with a high bandwidth requirement than would be allocated to data requiring less bandwidth). For example, if a 10 ms frame is assumed, 250 bytes will be allocated to line switched data requiring a bandwidth of 200K bps (facsimile) while only 10 bytes will be allocated to line switched data with an 8K bps bandwidth (voice). This technique provides both optimum utilization of line bandwidth and node flexibility with respect to available lines.

By requiring slots to be an integer number of bytes in length, the bandwidth inefficiency problems associated with the larger fixed-size slots are avoided. A large, fixed-size slot approach (e.g., 24 or 80 bits) will always waste some bandwidth in handling L/S calls with speeds not equal to exact multiples of the slot size. The reason for the wasted bandwidth is that the slot size is not small enough to be a common divisor for the various L/S channel speeds.

Because the time slots are allocated using a "best fit" algorithm, each call is located in the most optimum frame locations available at the time of call set up. There is no need to reorganize the frame format to obtain contiguous slots. As line switched data transmission is completed, the associated time slots become available for packet switched data. In Figure 5 the allocation and

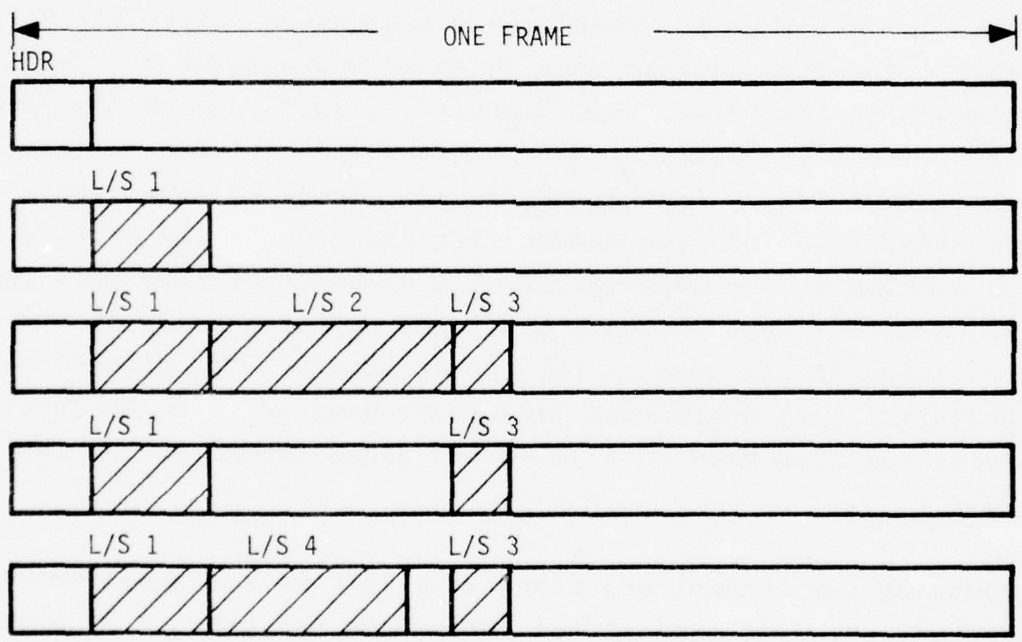


Figure 5. L/S Call Setup Takedown Example

deallocation of frame time slots are depicted. L/S calls one, two, and three are sequentially allocated. Then L/S call two is deallocated, and L/S call four is allocated according to the "best fit" algorithm.

Packet Data

Unlike the L/S data, the packet data (Classes IB, II, and III, is used to "fill in" whatever unused slots remain after L/S data locations have been allocated within the frame. A packet may range in size from 48 bits to approximately 2000 bits; hence, each packet data is used to "fill in" unused slots, a packet may be fragmented into noncontiguous slots.

Generally, there may be more than one packet in a frame. Each packet is delimited in front by a bit sequence (01111110) called a flag. The flag is also used as an idle character in the unused bandwidth of the frame. Bit stuffing is used to mask any appearance of the flag sequence that occurs within the packet itself. The bit stuffing is done at the transmit node by inserting a zero after a run of five consecutive ones, while the receive node removes a zero detected after a sequence of five ones (within a packet). This prevents the false representation of the flag by a string of six ones in the packet itself. The after-stuffed packets and flag delimiters (and flags used as idle characters, if any) are then used to fill in the slots unused by the L/S data.

Because system control and signalling information (call setup, takedown, etc.) is transmitted throughout the network as high priority packets, it is not possible to completely allocate the entire frame to line switched data. There will always be a

portion of each frame that will be reserved for this purpose (see Figure 4).

Packet switched data is transmitted in frame time slots that are not otherwise reserved (for Start-of-Frame marker (Frame header), Synchronization field, or line switched data). Placement of a packet into slots continues until the entire packet has been placed. Because packets are delimited by idle flags, they may be fragmented as necessary within a frame and may be continued from one frame to the next without incurring additional overhead (Figure 6).

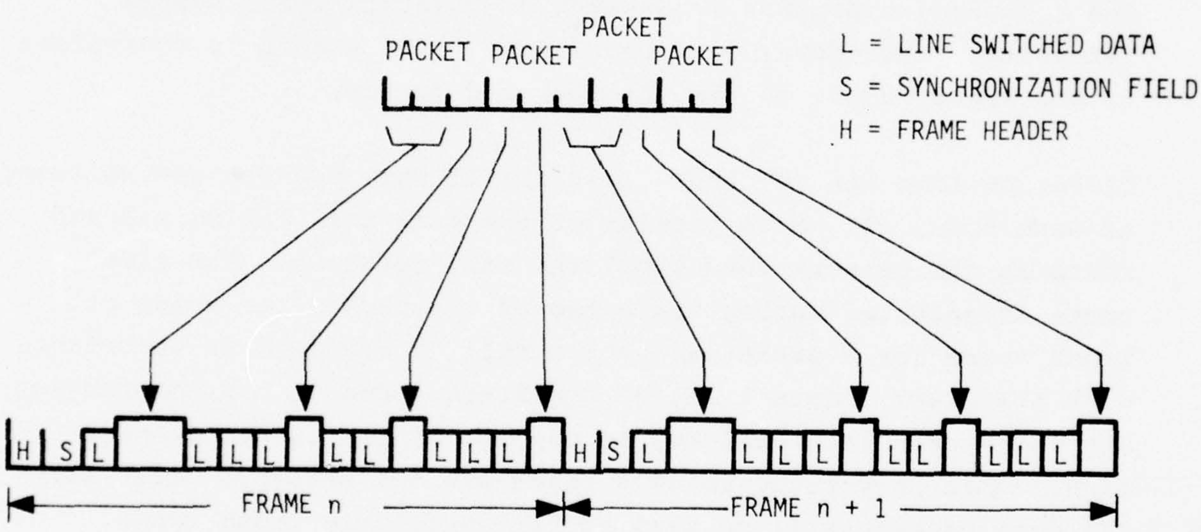


Figure 6. Packet Distribution

Packetization of Class IB data (clear voice) -- As discussed in the user requirements (Section 2), Class IB data are compressible and do not require bit count integrity (BCI) be maintained. After compression, data of a Class IB call may be characterized as having randomly distributed arrivals with an average rate of less than one-half the rate of its uncompressed equivalent. When transmitted, the size of each frame (10 ms) voice sample remains unchanged in the frame, and only a small amount of overhead is incurred during packetizing (voice packet format is described later in Section III). The overhead is 24 bits per packet, which has information content of 24 bits to 640 bits. The compression process is assumed to maintain frame sample integrity. Therefore, each packetized voice sample is equivalent to one frame sample of the uncompressed source.

Different from the way that Class II/III data packets are switched at each node, all voice packets of the same call follow a fixed route in the network throughout the call duration. The fixed route is selected during the setup of the packetized voice call. Trunk space for a packetized voice call is reserved in accordance with the Class I data limit on the frame. Due to the compression, a voice packet of a call may or may not appear in every frame time. When packetized voice samples are not present, Class II/III data packets will be used to fill available trunk space.

As voice packets are passed through the network, each node treats them as a special type of packet. But they are buffered in the same storage area as other packet types, and using the same packet processing facilities. The major differences between voice packet processing and regular Class II/III data packet processing are that: 1) voice packets of a call follow a predetermined fixed route while Class II/III data packets are dynamically

routed at each node; 2) voice packets are passed through the network without error checking; and, 3) voice packets have higher priority for transmission than any Class II/III data packet. The voice packet features cited in the differences 2) and 3) above minimize the delay or any skew of the voice samples through the network. Differences 1) and 2) are discussed further later in this section.

The effect of the packetized voice transmission method is to reduce the real-time bandwidth requirement for Class I data, thereby providing additional transmission capacity for Class II and III data.

LINK CONTROL

Communication between nodes is via a communication link. Nodes accomodate any of a variety of link types and speeds ranging from leased phone lines to satellite relay links. Each link is considered to be a pair of unidirectional lines, although pairing will not be a requirement. For purposes of discussion, only one link will exist between a pair of nodes. Obviously, multiple links are both possible and desirable. For each line, the sending node will be the line master and the receiving node will be the slave (Figure 7-A). The format and content placed on a line will be determined by transmit tables resident within the master. Interpretation and processing of a line's content will be determined by matching receive tables resident within the slave.

One example of a transmit table is shown in Figure 7-B. Each slot is described by one entry in the table. Each entry contains both the starting byte position of the slot and the

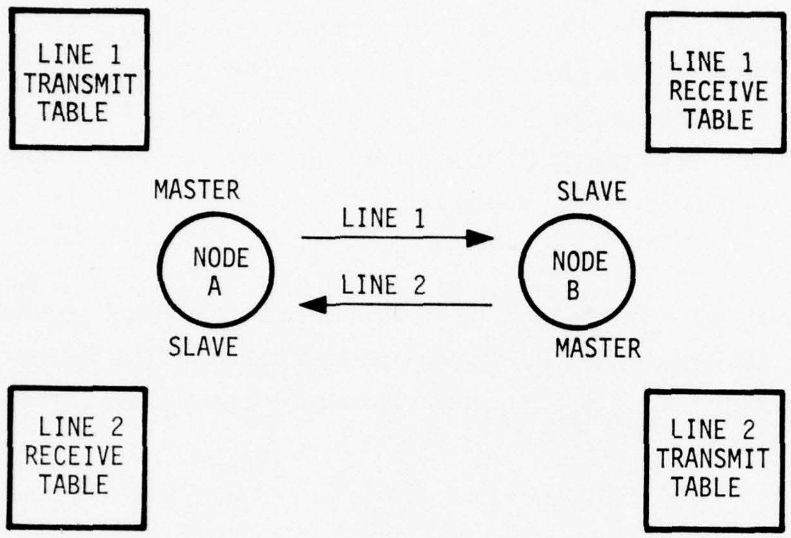


Figure 7-A. Network Link

	STATE	STARTING BYTE	SIZE	L/S CALL NO.	BUF NO.
P/S	00	25	1864		
L/S	11	4	6	87	211
P/S	00	20	2		
L/S	11	22	3	9	110
L/S	11	10	10	38	96

Figure 7-B. Transmit Table

size of the slot (in bytes). Use of the remaining fields is discussed later.

FRAME SYNCHRONIZATION ACQUISITION AND MAINTENANCE

Frame synchronization (sync) over a link between the transmitter of a node and the receiver of another node is necessary for the receiver to correctly interpret the data stream from the incoming link; that is, both the receiver and the transmitter must agree on the way the data stream on the link is interpreted. If frame sync is lost on a link, then the data transmitted over that link will be meaningless at the receiving end (i.e., lost) until the frame sync is reacquired. It is very desirable that the frame sync acquisition and reacquisition after a frame sync loss be very fast and that the probability of losing a frame sync be very small. It is also very desirable that a high transmission efficiency be achieved over the network links (i.e., the transmission bandwidth overhead due to acquiring and maintaining the frame sync should be kept to a minimum).

Two different approaches for frame sync acquisition and maintenance are described in the following sections. The first approach uses an 8-bit Start of Frame (SOF) marker in each 10ms frame. The second approach uses a longer (25-bit) SOF marker in each frame. The tradeoffs and comparisons between these two approaches will also be discussed.

Frame Synchronization Approach I

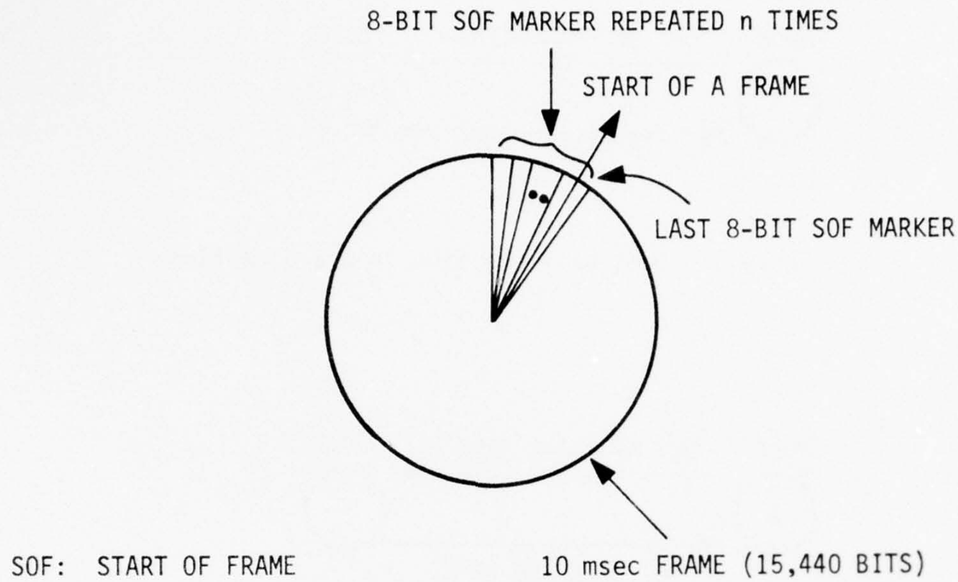
The 8-bit SOF marker used in Approach I contains a 7-bit Barker code sequence followed by a zero (11100100). This 8-bit SOF marker can always be uniquely identified among a 22-bit sequence

being composed of this SOF marker preceded by an arbitrary 7-bit sequence and followed by another arbitrary 7-bit sequence. This feature reduces the probability of incorrectly declaring frame sync at the receiving end. (An incorrect or false frame sync occurs when a receiver considers that the bit stream on the incoming link is in frame sync by matching SOF marker on data bits, but is in reality not in frame sync.)

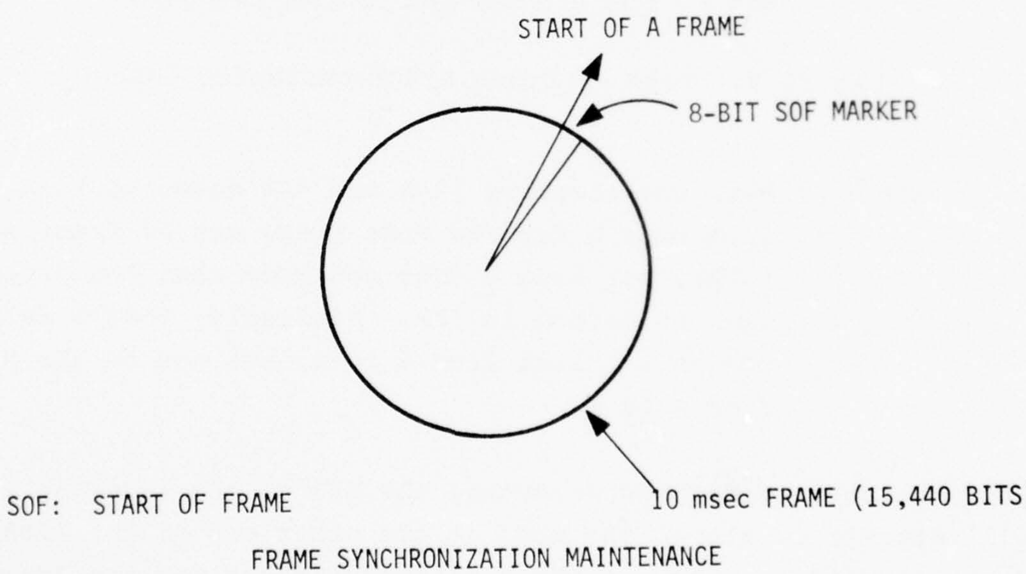
Acquisition of frame sync (initially or after a loss of frame sync) on a link is accomplished by commanding the transmitting end to send a number of consecutive 8-bit SOF markers to the receiving node. The number of SOF markers sent determines the certainty of the frame sync acquisition. The probability of frame sync acquisition at a wrong position (i.e., an arbitrary data sequence matches the sequence of the series of 8-bit SOF markers) is 2^{-8n}, where n is the number of 8-bit SOF markers being sent. The information bits for the first frame follow the last of the n 8-bit SOF markers, and the frame sync is established thereafter. That is, an 8-bit SOF marker will appear in every 10 msec after the last 8-bit SOF marker (see Figure 8). Once the frame sync is established, the frame sync can be maintained reliably due to the unique feature of the 7-bit Barker code contained in the SOF marker. The SOF marker is predicted and and checked by the receiving end at the start of every 10 msec frame cycle.

To reacquire frame sync after a node has detected a loss of frame sync on the incoming data stream, two cases will be considered (refer to Figure 9):

Case 1 At Node B, the incoming link from Node A has lost frame sync but the outgoing link from Node B to Node A is still in frame sync.

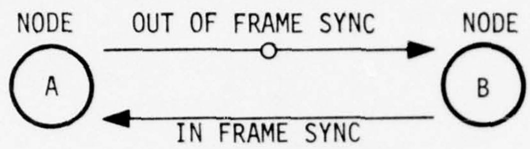


Frame Synchronization Acquisition/Reacquisition

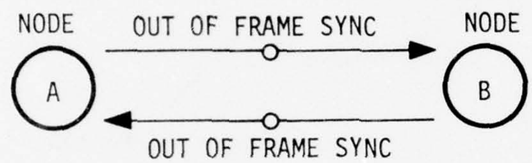


Frame Synchronization Maintenance

Figure 8. Frame Synchronization Approach I



Case 1: Loss of Frame Sync in One Direction



Case 2: Loss of Frame Sync in Both Directions

Figure 9. Loss of Frame Synchronization Cases

Case 2 Both the incoming link and the outgoing link to/from Node B from/to Node A are out of frame sync (OFS), but Node B does not know that the outgoing link to Node A is OFS. Similarly, Node A detects OFS on the link from B to A, but not on the Node from A to B.

In both cases, each node detecting the OFS on the incoming link will attempt to signal the node at the other end of the link to start to send a number of the 8-bit SOF markers for the frame sync reacquisition on the incoming link. One way of signalling the OFS is to complement each bit of the 8-bit SOF marker for the next outgoing frame. For Case 1, since the link from Node B to

Node A is still in frame sync, the signal received at Node A will be correctly interpreted and Node A can start to send a number of the SOF markers to Node B to allow Node B to reacquire the frame synchronization for the link from A to B.

For Case 2, both Nodes A and B will attempt to signal the opposite side about the OFS states, but neither of them can get the message across because the links for both directions are out of frame sync. To solve this problem, a "Time-Out" procedure should be used at each node that sends the OFS signal. That is, after a predetermined time, if a node does not start to receive a series of SOF markers after it has attempted to signal the opposite node about the OFS status, the node assumes that the outgoing link is also OFS. It then starts to send a series of SOF markers to the opposite end. In Case 2, the Time-Out will occur at both Nodes A and B, while in Case 1, the Time-Out will not occur. With the provision of the Time-Out technique, in both Case 1 and 2, the frame sync for the links can be acquired properly.

In Case 1, the frame sync reacquisition can be accomplished in less than two-frame times (a maximum of one frame time for Node B waiting to send OFS signal to Node A and a fraction of one frame time for Node A sending a number of the 8-bit SOF markers), plus the transmission time on the links (from B to A and then from A to B). For Case 2, the frame sync reacquisition can be accomplished in one Time-Out and a fraction of one frame time (for sending the SOF markers) plus the transmission time on one link. Since the Time-Out period is close to (or slightly less than) the time required for Case 1 frame sync reacquisition, the frame sync reacquisition for Case 2 can be accomplished in a time that is not much longer than that for Case 1.

The above discussions on the frame sync reacquisition time assume that there was no error on the transmission line during the transmission of the OFS signal and the series of SOF markers. However, if there is an error in transmitting the signal, or the SOF markers during reacquiring the frame sync, the receiving end will not receive the expected series of SOF markers. The frame sync reacquisition procedure has to be restarted after another Time-Out. If the number of 8-bit SOF markers is large (e.g., 15) and the transmission line has a rather high bit error rate (e.g., 10^{-3}), then there exists a high probability (0.12) that some error occurs among the SOF markers (total 120 bits), and a retry of the frame sync reacquisition is required. In such cases, the frame sync reacquisition time will be longer than that described in the previous paragraph.

Frame Synchronization Approach II

The second frame sync approach employs a longer (25 bits) Start of Frame (SOF) marker which is sent once every 10-msec frame. The SOF marker is composed of one 11-bit Barker code preceded and followed by a 7-bit Barker code at each end (1110010 11100010010 1110010). This 25-bit sequence can always be uniquely identified among a 59-bit pattern containing the 25-bit sequence preceded and followed by any 17-bit pattern. As a result, the probability of false frame synchronization will be very small.

The second frame sync approach differs from the first approach primarily in the frame sync acquisition/reacquisition procedure. In the second approach to achieve a frame sync, the 25-bit SOF marker is sent only once from the transmitter of one node to the receiver of another node at the opposite end of the link. (The first approach sends n 8-bit SOF markers.) Once the receiver

detects the 25-bit SOF marker, the frame sync is established. (The information bits for the first frame follow the 25-bit SOF marker.) The error probability of the frame sync acquisition is 2^{-25} (or 2.98×10^{-8}). After the frame sync is established, every 10 msec thereafter the receiver predicts and checks the 25-bit SOF marker which indicates the beginning of each frame (see Figure 10).

When a receiver checks the predicted bit positions for the 25-bit SOF marker on the incoming bit stream of a link and fails to match the SOF marker sequence pattern, the receiver reverts to an OFS mode and starts to hunt for the SOF marker pattern on the incoming bit stream. Once a 25-bit pattern matches the SOF marker, the receiver re-establishes frame sync. (The bits following the matching 25-bit are considered as valid data of the first frame after recovery.)

Using this approach, the receiver does not need to send a signal to the transmitter to indicate the OFS status. The time required to get back into sync after loss of a frame sync is normally less than one frame time and thus is shorter than that required by the first approach.

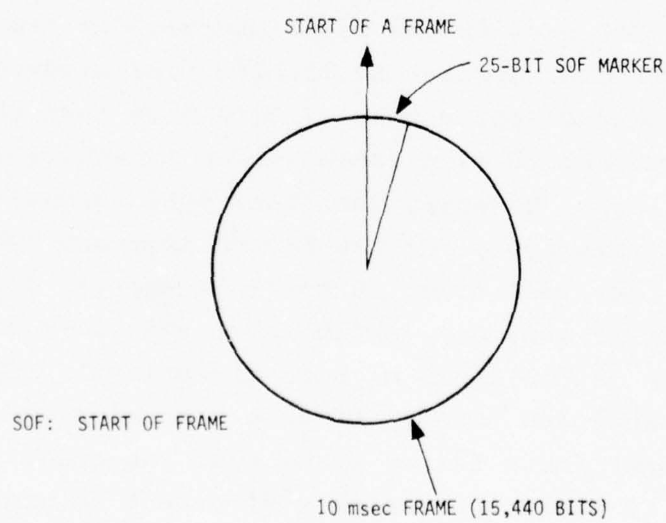


Figure 10. Frame Synchronization Approach II

The 25-bit SOF marker is sent once every 10 msec frame (assume 15,440 bits). In the process of reacquiring the frame sync after loss of a frame sync, the probability that any 25-bit sequence among the 15,440 bit other than the actual SOF marker matches the 25-bit SOF marker is:

$$1 - \left(1 - \frac{1}{2^5}\right)^{15,440-49} \times \left(1 - \frac{1}{2^{18}}\right)^2 = 4.67 \times 10^{-4}$$

This is the probability that a frame sync reacquisition may assume SOF markers at a wrong position. As shown, this probability is very low. In case this happens, the wrong frame sync can be readily detected, one frame later, by checking the SOF marker at the assumed SOF position. A normal frame sync reacquisition procedure as described earlier will then be restarted.

Comparison of the Two Frame Sync Approaches

Each of the two frame sync approaches employs a Start of Frame (SOF) marker which is predicted and checked at the beginning of each 10 msec frame. The SOF used in the first approach is 8-bit, while the SOF used in the second approach is 25-bit. The bandwidth overhead incurred by the second approach is 17 bits more than that by the first approach in each 10 msec frame (which is equivalent to 15,440 bits on the T1 line). However, the frame sync reacquisition time, after loss of a frame sync, for the second approach (which normally takes less than one frame time, 10 msec) is shorter than that required for the first approach (which takes two frame times plus the transmission time on the links of both directions). The point is that the second approach trades off some transmission line bandwidth (17 bits per frame) for a faster frame sync reacquisition after loss of frame sync. If the transmission environment is such that the loss of a frame sync seldom occurs, then the total frame sync reacquisition

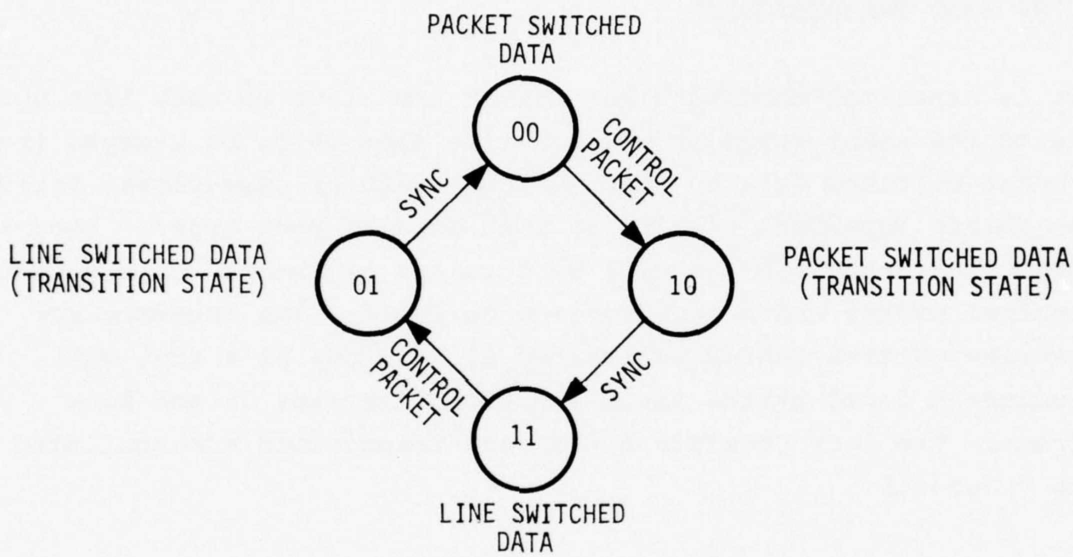
time saved by the second approach may be insignificant and not worth the 17-bit bandwidth overhead spent in each frame. A qualitative conclusion is that the first frame sync approach is more suitable for a network with low bit error rates on the transmission lines (and hence loss of frame sync seldom occurs), while the second approach is more appropriate otherwise.

TIME SLOT TRANSITIONING

It is essential that both the master and slave on each line agree as to the exact frame in which a time slot is to be changed from packet-switched data to line-switched data or vice-versa. Failure to obtain agreement results in lost data of both types. Accurate time slot transitioning will be obtained by the use of a network control packet and a synchronization field. The transmit and receive control tables maintained at the ends of a line will include a 2-bit status field for each time slot in the line frame. The four possible states and transitions are indicated in Figure 11.

As each new line-switched call is established, the line master reserves one or more "best-fit" time slots to provide the required bandwidth. It then sends a control packet describing those slots to the line slave. An identifying number called the node-to-node call setup number is included in the control packet. Only after reservation of the entire call path through the network is completed can the actual transition be initiated. Transmission of a valid synchronization field containing the node-to-node call setup number effects the actual transition. Because of its relatively short length, the synchronization field can be easily encoded with a powerful error correcting capability. The field contains a flag bit which alerts the slave node to the fact

that slot transitions are occurring. Upon detection of the synchronization flag bit the node extracts the node-to-node call setup number and searches its receive table to locate those time slots which correspond to the new call. The appropriate status fields are then updated to indicate line switched data. Frame decomposition can then proceed without error.



TIME SLOT STATUS STATES

STATE	MEANING
00	PACKET SWITCHED DATA
10	PACKET SWITCHED DATA (TRANSITION TO LINE SWITCHED DATA PENDING)
11	LINE SWITCHED DATA
01	LINE SWITCHED DATA (TRANSITION TO PACKET SWITCHED DATA PENDING)

Figure 11. Time Slot Transitions

The transition from line-switched data to packet-switched data occurs in much the same manner, except that: 1) only the call setup number is transmitted to the slave node; and 2) different status states are involved.

PACKET TYPES AND FORMATS

All classes of user data except Class IA are transmitted over the network in packet form, In addition, system control and signalling information (call setup, takedown, etc.) is also transmitted throughout the network as packets. Categorically, packet switched data are divided into the following four types:

- Data packets (Class II/III user data)
- Control packets (For Class IA/IB call setup, takedown, or for Class II/III data transmission procedures)
- Acknowledge (Ack) packets
- Voice packets (Class IB--clear voice--call 10-ms samples)

Figure 12 shows the packet format for data packets and control packets. Figure 13 shows the packet format for Ack packets, and Figure 14 shows the packet format for voice packets. As discussed earlier in the network frame format description, an 8-bit flag (01111110) always precedes a packet. The 8-bit flag indicates the beginning of a packet. Each packet has a 4-bit packet-type field (in packet header) which indicates a particular packet type to which this packet belongs.

Packet Format for Data Packet or Control Packet (Figure 12)

Each data or control packet has a packet header field and a text field, followed by a Cyclic Redundancy Check field. These fields

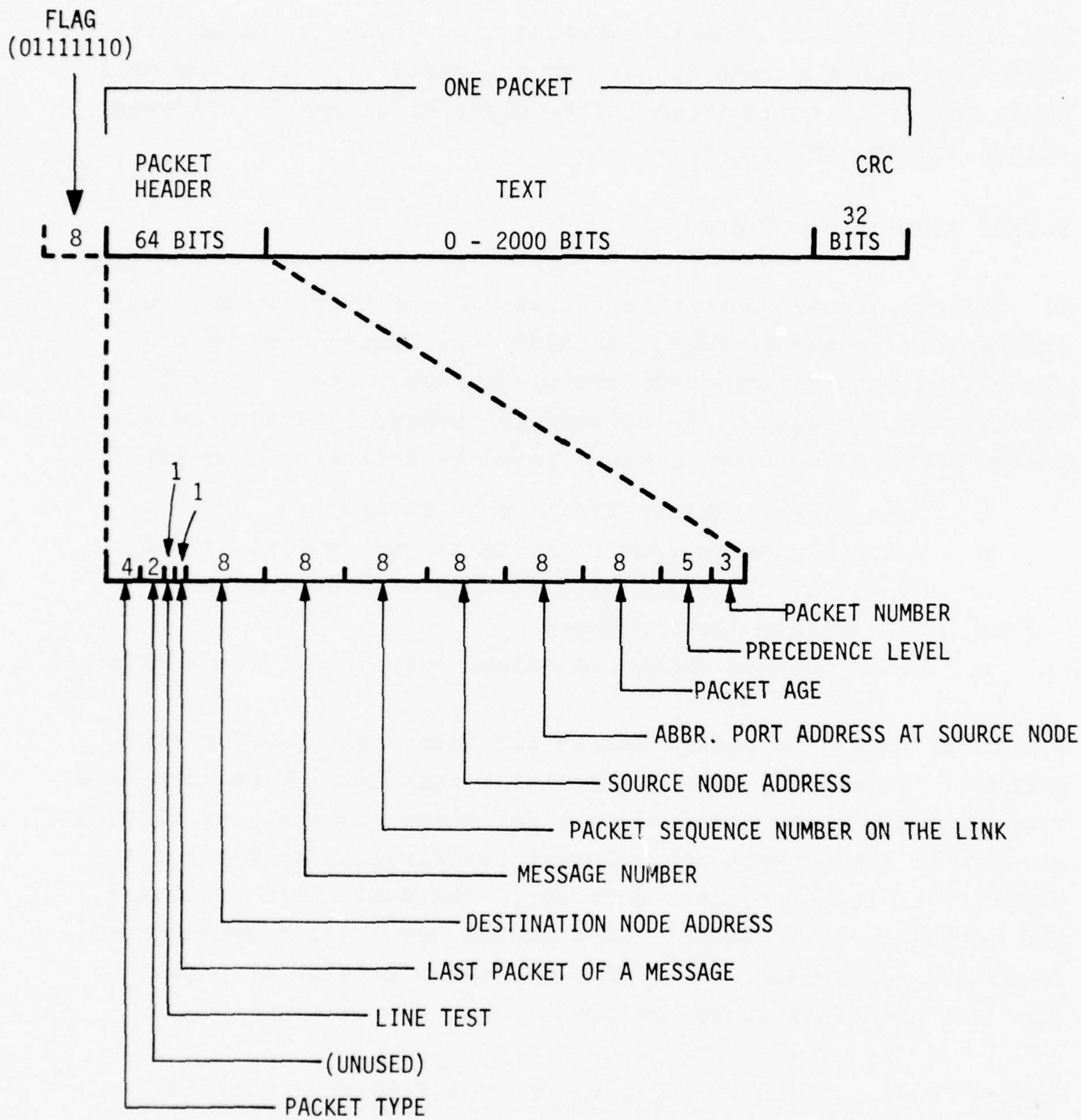


Figure 12. Packet Format for Data or Control Packet

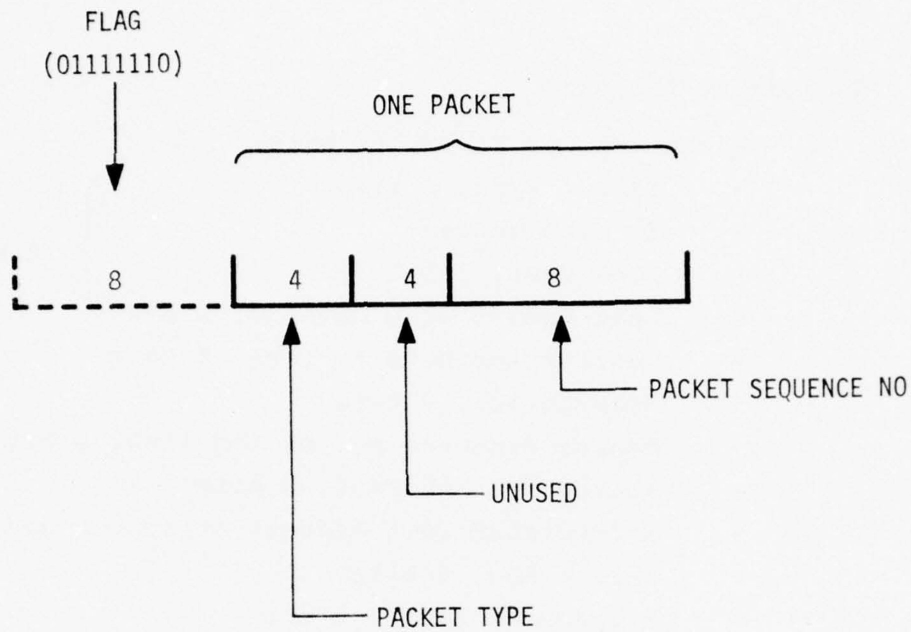


Figure 13. Packet Format for Acknowledge Packet

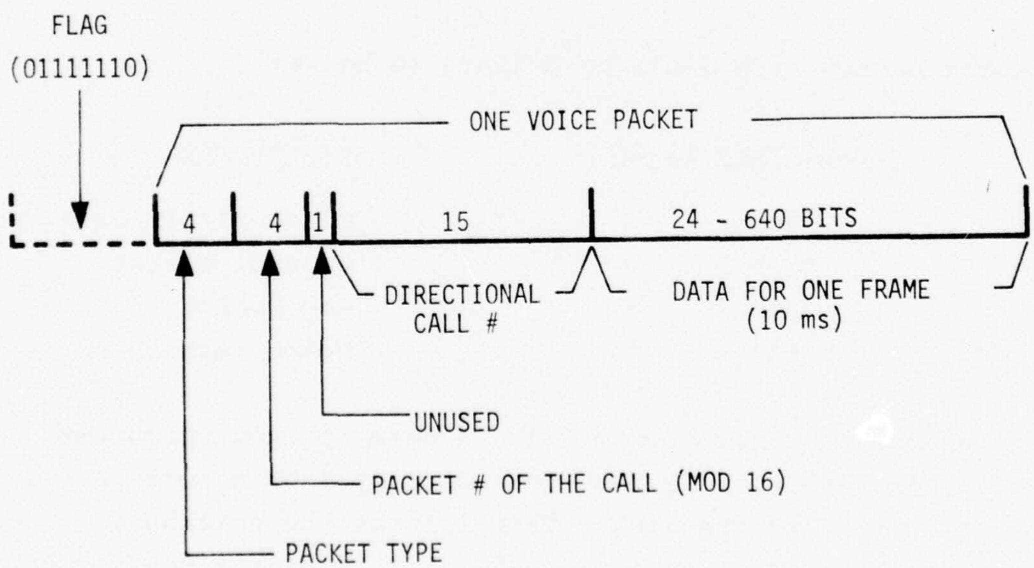


Figure 14. Voice Packet Format

are defined as follows:

- Packet Header Field, 64 bits:
  - Packet type, 4 bits
  - (2 bits unused)
  - Line Test, 1 bit
  - Last Packet of a Message, 1 bit

  8 bits
  - Destination Node Address, 8 bits
  - Message No., 8 bits
  - Packet Sequence No. on the link, 8 bits
  - Source Node Address, 8 bits
  - Abbreviated Port Address at source node, 8 bits
  - Packet Age, 8 bits
  - Precedence Level, 5 bits
  - Packet No. (within a message), 3 bits
- Text Field, 0-250 8-bit bytes
- Cyclic Redundancy Check field (CRC), 32 bits

The 4-bit packet type field is defined as below:

Packet Type Field	Packet Type
1000	Class II/III Data Packet
0100	Control Packet
0010	Ack Packet
0001	Voice Packet

When there is no traffic on a link, a dummy packet which has "Line Test"bit set can be sent periodically from a node to its neighboring node of the link. This informs the neighboring node about the line status (good or failure) so that its routing table can be updated accordingly.

A bulk of Class II/III data to be transmitted over the network has to be divided into units of messages where each message cannot contain more than 16,000 bits. The 8-bit Message No. field is used to indicate the order of the messages from the same bulk data source.

A message is packetized before being transmitted over the network, and a message may be transmitted by up to eight data packets. These packets are numbered from zero up to seven (Packet No.). The last packet in a message is indicated by having its "Last Packet of a Message" bit set.

The "Packet Sequence Number on the Link" is used in the node-to-node protocol for transmitting data packets and control packets from one node (sending node) to an adjacent node (refer to "Node-to-node Packet Protocol" on page 104). It is used to relate acknowledgments to transmitted data and control packets to the sending node, so that successful transmission of these packets is assured before their security copies can be deleted at the sending node. The Packet Sequence Number is in sequential order and it is a number modulo 256.

The Source Node Address designates the node which originates the packet (where a message is packetized), and the Destination Node Address designates the node to which the packet is destined (where packets of the same message are reassembled into a message). Assuming that the network has at most 64 nodes, 6 bits will be sufficient for the node addresses. However, 8 bits are used, so that each node address fits into one byte.

The Abbreviated Port Address at Source Node is a "during-the-call" local port address, which is assigned by the source node during the setup of a user call. The 8-bit Abbreviated Port address allows

up to 256 users from a node to be source ports to the network at the same time for each of the Class IA, Class IB, and Class II/III user types. This is because Class IA data, Class IB data, and Class II/III data are handled as three completely separate categories in the network. The limitation of 256 source ports at a node does not restrict the number of destination ports at that node.

The 8-bit Packet Age indicates the age of the packet in the network since it was packetized at the source node. The 5-bit Precedence Level contained in a data packet (Class II/III) indicates the precedence level of the packet which is the same as the user precedence level. Both parameters are for determining the order of routing processing for this packet and the order of the packet to be placed in an output trunk (see "L/S and P/S Routing Considerations" on page 107. The Precedence Level contained in a control packet is the user precedence level (not the precedence level of the control packet) for the user whose call the control packet attempts to affect (e.g., setup, takedown, transmission, procedure control, etc.). Control packets have the highest priority for transmission among all packets. Their packet type, rather than their precedence level field, is consulted in determining their order of being routed to and placed in an output trunk.

The Text field is for the information content. It may have from 0 to 250 8-bit bytes (i.e., 2000 bits maximum).

The 32-bit Cyclic Redundancy Check (CRC) field is for error detection for the content of the complete packet after it is received at a node.

Packet Format for Acknowledge (Ack) Packet (Figure 13)

An Ack packet has two bytes. The first byte contains a four-bit field for the Packet Type and another four bits unused. The second byte is the Packet Sequence Number on the Link. It has the same sequence number as the packet (data or control) to which the Ack packet acknowledges correct acceptance at a receiving node. The Ack packets are used only for node-to-node protocol for data and control packets. There is no CRC field in an Ack packet. There is no error checking or acknowledgement for an Ack packet.

Packet Format for Voice Packet (Figure 14)

A voice packet (for 10 msec data of clear voice channel) has the following fields:

- Packet Type, 4 bits
- Packet Number of the call (modulo 16), 4 bits
- 1 bit unused
- Directional Call Number for the packet, 15 bits
  - Direction of the packet to or from called node, 1 bit
  - Call Number, 14 bits
    (Source Node Address, 6 bits)
    (Abbreviated Source Port Address, 8 bits)
- Clear Voice Sample for 10 msec of the call (in one direction), 24-640 bits (3-80 bytes)

The four-bit packet type field distinquishes voice packets from other packet types (Class II/III data packets, control packets, Ack packets). The packet number field (4 bits) is used at the destination node for voice reconstruction. The directional call number uniquely identifies the output trunk for each frame sample packet of the call, and thereby allows the network to transmit each frame sample packet to its proper destination. There is no CRC field in a voice packet. There is no error checking or acknowledgement for a voice packet.

Note that regardless of the packet type, the length of any packet is a multiple of 8-bit bytes. Therefore, the packet formats are also adaptable for the other technique of delimiting packets which employs "byte stuffing" method (i.e., using 8-bit control characters such as DLE, STX, ETX, etc. for packet delimiters as in the ARPA Network).

CONTROL PACKETS

In the proposed integrated network, the primary method of transferring control information between nodes is via control packets. Control packets have the highest priority for placement on a transmission trunk. There are 13 different types of control packets.

There are five types of control packets for handling L/S call setup and takedown:

- L/S slot reservation command/request (Type 1)
- L/S slot reservation command (Type 1a)
- L/S slot reservation request denied (Type 2)
- L/S slot unreserve command (Type 3)
- L/S slot de-reservation command (Type 4)

There are five types of control packets for handling P/S transmission procedures to facilitate the efficient transmission of single and multi-packet messages through the network:

- P/S transmission connection request (Type 5)
- P/S transmission connection answer (Type 6)
- Multi-packet transmission request (Type 7)
- Ready to receive multi-packet message (Type 8)
- Confirm acceptance of a single-packet message (Type 9)

There are three types of control packets for handling packetized voice (P/V) call setup and takedown:

- P/V call setup request/command (Type 10)
- P/V call setup request denied (Type 11)
- P/V call takedown command (Type 12)

The use of control packets and the nodal processing after a node receives a certain type of control packet will be described in the following sections on L/S call setup/takedown, P/V call setup/take-down, and P/S data transmission procedures. The content and the size of each of the control packets and their definitions are given in Appendix A.

LINE-SWITCHED CALL SETUP/TAKEDOWN PROCEDURES

Introduction

The line-switched call setup (takedown) procedures consist of two steps: 1) slot reservation (de-reservation), and 2) allocation (de-allocation).

The first step is concerned with slot reservation (de-reservation). Its purpose is to prime the selected nodes (based on route selected) for the transition from (to) packet-switched data to (from) line-switched data. Reservation (de-reservation) is accomplished through the transmission of specialized control packets. Control packets used for reservation identify the time slots that will be used by the call being set up. Control packets used for de-reservation identify a line-switched call by its call setup number, which is in turn linked to the time slots used for the call. (Control packet details are described later.) At the completion of reservation (de-reservation) the nodes are ready for the second step, allocation (de-allocation).

The second step, concerned with the allocation (de-allocation) of time slots, is accomplished through the transmission of a valid synchronization field (refer to frame format description). The synchronization field specifies the call, and thereby the time slots, which are to be allocated (de-allocated). At the completion of allocation (de-allocation), the nodes are transmitting (not transmitting) line-switched data in the selected time slots.

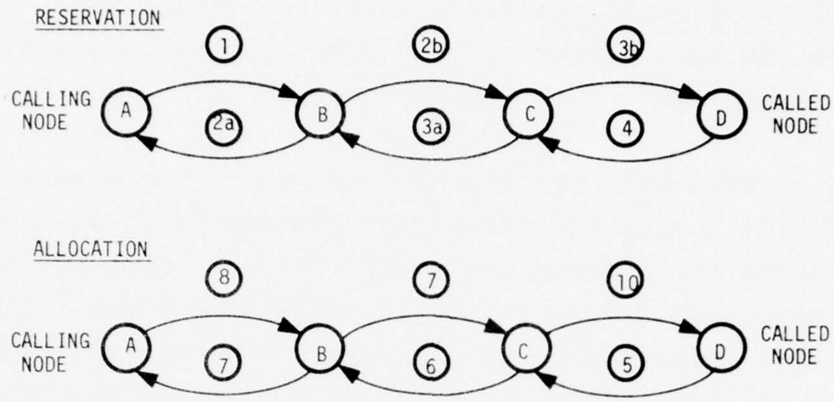
The line-switched call setup (takedown) procedures cause the transmit-and-receive-tables internal to the network nodes to be updated. Reservation causes time slots containing packet-switched data (state 00) to be designated as packet-switched data with transition pending (state 10). Allocation is the transition process and causes the states of the time slots to be updated from 10 to 11 (line-switched data). De-reservation again results in slots being designated as transition pending (state 01), and de-allocation is the transition process to packet-switched data (state 00).

The line-switched procedures discussed here include setup, takedown, and preemption.

Normal Line-Switched Call Setup

Reservation -- Line-switched call setup begins when the calling node, A (Figure 15) responds to one of its local subscribers by sending a "L/S slot reservation command/request" (Control Packet Type 1) to the next node, B. This begins reservation of the forward path.

The second node, B, responds by sending a "L/S slot reservation command" (Control Packet Type 1a) back to A. This begins reservation of the return path. Node B also continues the forward reservation by sending a Type 1 control packet to node C once it has been determined that the return path from B to A can be reserved.



NODE A CALLS NODE D

PACKETS

1 RESERVE LINE/SLOT FROM A TO B
(TYPE 1 CONTROL PACKET)

2a RESERVE LINE/SLOT FROM B TO A
(TYPE 1a CONTROL PACKET) <ACKNOWLEDGE NOT SHOWN>

2b RESERVE LINE/SHOT FROM B TO C
(TYPE 1 CONTROL PACKET)

3a RESERVE LINE/SHOT FROM C TO B
(TYPE 1a CONTROL PACKET) <ACKNOWLEDGE NOT SHOWN>

3b RESERVE LINE/SLOT FROM C TO D
(TYPE 1 CONTROL PACKET)

4 RESERVE LINE/SLOT FROM D TO C
(TYPE 1a CONTROL PACKET) <ACKNOWLEDGE NOT SHOWN>

FRAMES

5 ALLOCATE RESERVED D TO C LINE/SLOT (SYNC)

6 ALLOCATE RESERVED C TO B LINE/SLOT (SYNC)

7 ALLOCATE RESERVED B TO A LINE/SLOT (SYNC)

8 ALLOCATE RESERVED A TO B LINE/SLOT (SYNC)

9 ALLOCATE RESERVED B TO C LINE/SLOT (SYNC)

10 ALLOCATE RESERVED C TO D LINE/SLOT (SYNC)

Figure 15. Normal Line Switched Call Setup Procedure

The reservation process continues until the called node, D, has sent and received an acknowledge for the Type 1a control packet which completes reservation of the return path.

Allocation -- When both the forward and return paths have been reserved, the called node begins the allocation process by 1) sending a valid synchronization field along the return path (to node C); and 2) including line-switched data in the reserved time slots. The synchronization field is transferred through each node at frame speed until it reaches the calling node. There its direction is reversed and line switched data is supplied in the next frame for the time slots reserved for the forward direction. When the synchronization field reaches the called node, allocation is complete (and the L/S data for the first frame in each direction has also been transmitted through the network.)

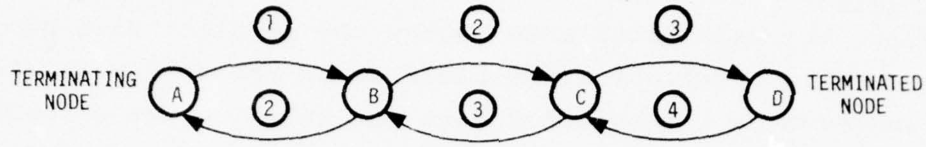
Normal Line-Switched Call Takedown

De-reservation -- Line-switched call takedown begins when the terminating node, A (Figure 16) responds to one of its local subscribers by sending a "L/S slot de-reservation command" (Control Packet Type 4) to the next node, B. This begins de-reservation of one direction of the call path.

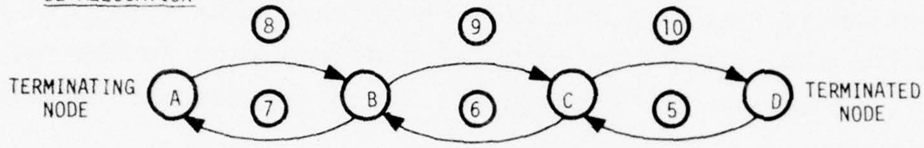
The second node, B, responds by sending a "L/S slot de-reservation command" back to A. This begins de-reservation of the second direction of the call path. Node B also continues de-reservation in the first direction by sending a Type 4 control packet to node C.

The de-reservation process continues until the terminated node, D, has sent and received an acknowledge for the Type 4 control packet which completes de-reservation of the second direction of the call path.

DE-RESERVATION



DE-ALLOCATION



NODE A TERMINATES

PACKETS

1 DE-RESERVE LINE/SLOT FROM A TO B
(TYPE 4 CONTROL PACKET)

2 DE-RESERVE LINE/SLOT FROM B TO A
(TYPE 4 CONTROL PACKET) <ACKNOWLEDGE NOT SHOWN>

DE-RESERVE LINE/SLOT FROM B TO C
(TYPE 4 CONTROL PACKET)

3 DE-RESERVE LINE/SLOT FROM C TO B
(TYPE 4 CONTROL PACKET) <ACKNOWLEDGE NOT SHOWN>

DE-RESERVE LINE/SLOT FROM C TO D
(TYPE 4 CONTROL PACKET)

4 DE-RESERVE LINE/SLOT FROM D TO C
(TYPE 4 CONTROL PACKET) <ACKNOWLEDGE NOT SHOWN>

FRAMES

5 DE-ALLOCATE D TO C LINE/SLOT (SYNC)
6 DE-ALLOCATE C TO B LINE/SLOT (SYNC)
7 DE-ALLOCATE B TO A LINE/SLOT (SYNC)
8 DE-ALLOCATE A TO B LINE/SLOT (SYNC)
9 DE-ALLOCATE B TO C LINE/SLOT (SYNC)
10 DE-ALLOCATE C TO D LINE/SLOT (SYNC)

Figure 16. Normal Line Switched Call Takedown

De-allocation -- When both directions of the call path have been de-reserved, the terminated node begins the de-allocation process: it sends a valid synchronization field along the call path (to node C) and; it includes packet-switched data (or idle) in the de-reserved time slots. The synchronization field is transferred through each node at frame speed until it reaches the terminating node. There its direction is reversed and packet-switched data (or idle) is supplied for the time slots which were de-reserved in the opposite direction. When the synchronization field reaches the terminated node, de-allocation is complete.

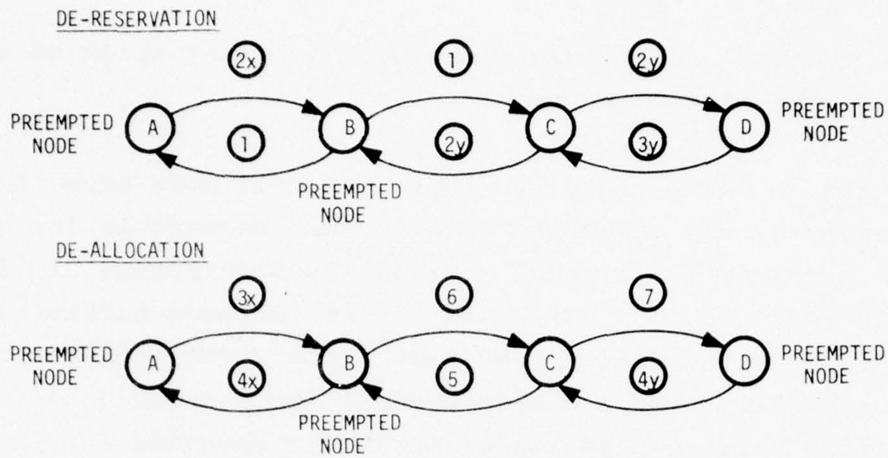
Line-Switched Call Preemption

Line-switched call preemption occurs when a network node elects to terminate a call to provide a route for a higher precedence call. Preemption by the calling or the called node is the same as call takedown. Preemption by an intermediate node is analogous to the simultaneous takedown of two calls.

De-reservation -- Line-switched call preemption begins when the preempting node, B (Figure 17), sends a "L/S slot de-reservation command" (Control Packet Type 4) to the adjacent nodes, A and C. This begins de-reservation of one direction on each of two call path segments.

The adjacent nodes, A and C, respond by sending a "L/S slot de-reservation command" back to B. This begins de-reservation of the second direction of the call path segments. The adjacent nodes may, as in the case of node C, continue de-reservation by sending a Type 4 control packet along the call path.

The de-reservation of a call path segment continues until the preempted node on that segment has sent and received an acknowledge for



LINE/SLOT FROM B TO C PREEMPTED BY NODE B

PACKETS

1 DE-RESERVE LINE/SLOT FROM B TO C (TYPE 4 CONTROL PACKET)

DE-RESERVE LINE/SLOT FROM B TO A (TYPE 4 CONTROL PACKET)

2y DE-RESERVE LINE/SLOT FROM C TO B (TYPE 4 CONTROL PACKET) <ACKNOWLEDGE NOT SHOWN>

DE-RESERVE LINE/SLOT FROM C TO D (TYPE 4 CONTROL PACKET)

2x DE-RESERVE LINE/SLOT FROM A TO B (TYPE 4 CONTROL PACKET) <ACKNOWLEDGE NOT SHOWN>

3y DE-RESERVE LINE/SLOT FROM D TO C (TYPE 4 CONTROL PACKET) <ACKNOWLEDGE NOT SHOWN>

FRAMES

3x DE-ALLOCATE A TO B LINE/SLOT (SYNC)

4y DE-ALLOCATE D TO C LINE/SLOT (SYNC)

4x DE-ALLOCATE B TO A LINE/SLOT (SYNC)

5 DE-ALLOCATE C TO B LINE/SLOT (SYNC)

6 DE-ALLOCATE B TO C LINE/SLOT (SYNC)

7 DE-ALLOCATE C TO D LINE/SLOT (SYNC)

Figure 17. Line Switched Call Preemption

the Type 4 control packet which completes de-reservation of second direction of the call path segment.

De-allocation -- When both directions of a call path segment have been de-reserved, the preempted node on that segment begins the de-allocation process: 1) it sends a valid synchronization field along the segment (D to C, or A to B) and; 2) it includes packet-switched data (or idle) in the de-reserved time slots. The synchronization field is transferred through each node at frame speed until it reaches the preempting node. There its direction is reversed and packet-switched data (or idle) is supplied for the time slots which were de-reserved in the new direction. When the synchronization field reaches the preempted node, de-allocation is complete.

Timing and Control -- Note that because the two call path segments are de-reserved and de-allocated independently, de-allocation can begin on one segment before de-reservation is complete on the other segment. Note also that, if necessary, nodes can cause preemption of line/slots of which they are not the master by preempting the corresponding line of the call path. For example, node B can cause preemption of the line from C to B by preempting the same call on the line from B to C.

L/S Node Processes and Process Flow Sequences

To appreciate and analyze the processing requirements created at a node by the L/S call handling procedures, a set of node processes and a set of process flow sequences have been defined. The process flow sequences are keyed to the receipt of control packets.

Node Processes -- The eight node processes are functionally defined:

1. Frame decomposer
2. Control packet processor
3. Synchronization field processor
4. Control packet formatter
5. Receive table updater
6. Transmit table updater
7. L/S router
8. Frame composer

Frame Decomposer -- The frame decomposer (Figure 18) separates incoming frames into:

- Synchronization field
- L/S data
- Control packets
- Data packets
- Voice packets

Each item is placed into the appropriate queue for processing. Recognition and separation of control, voice, and data packets may be performed by a subordinate process. The frame decomposer receives input from:

- The incoming communications line
- The receive table
- Node control

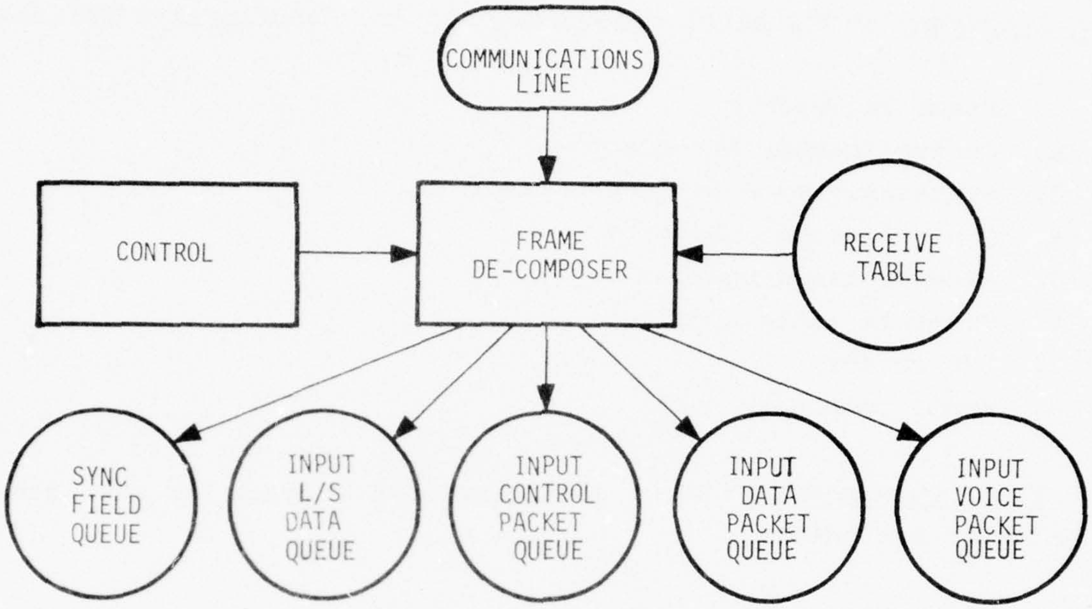


Figure 18. Frame Decomposer

Control Packet Processor -- The control packet processor (Figure 19) serves the control packet queue. It interprets control packets and appropriately requests the following actions:

- Receive table updates
- Transmit table updates
- Route selection

Each request is placed in the proper table update or route request queue.

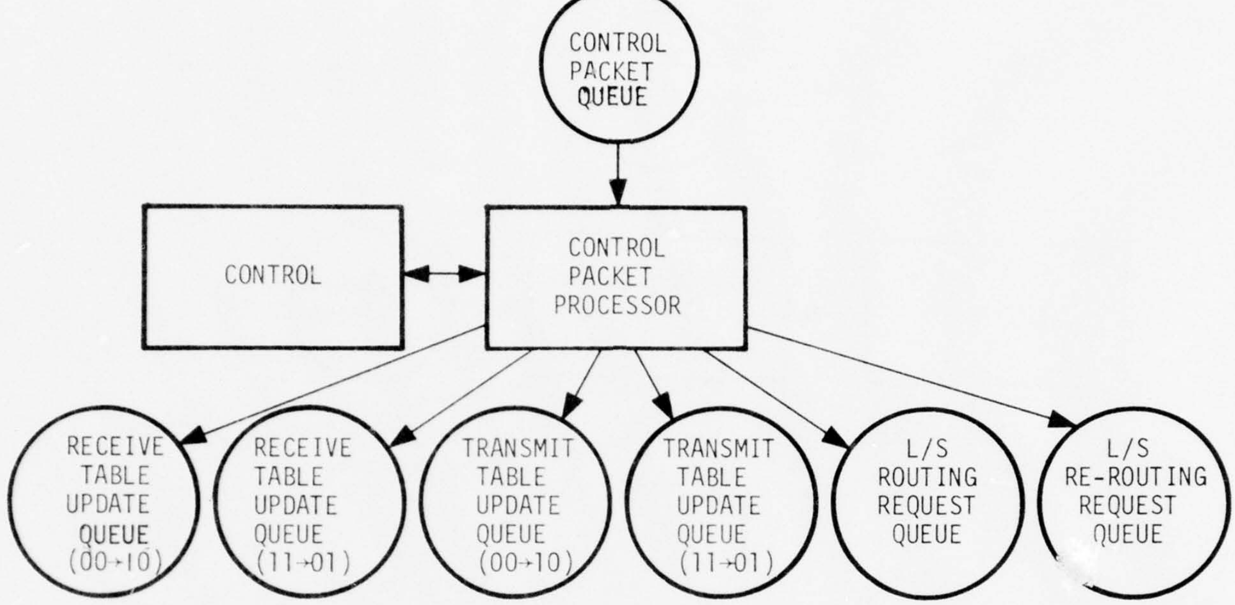


Figure 19. Control Packet Processor

Synchronization Field Processor -- The synchronization field processor (Figure 20) serves the sync field queue. It responds to valid synchronization field occurrences by requesting the following actions:

- Receive table state updates
- Transmit table state updates
- Synchronization field transmission

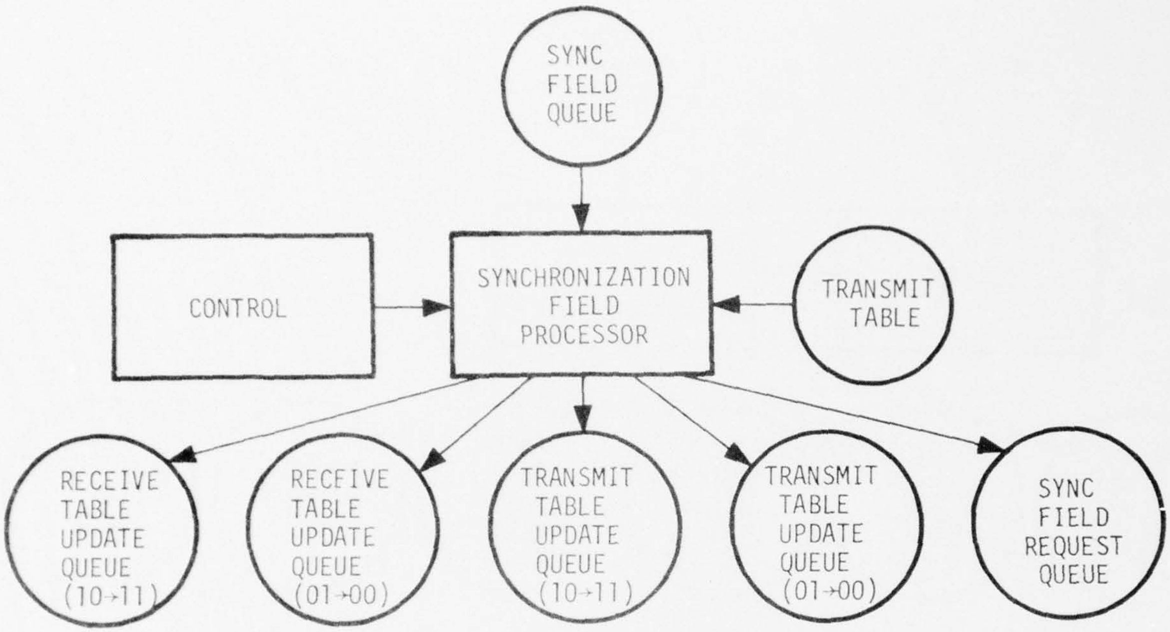


Figure 20. Synchronization Field Processor

Control Packet Formatter -- The control packet formatter (Figure 21) serves node control by creating control packets for transmission to other network nodes. The control packets are placed in the output control packet queue.

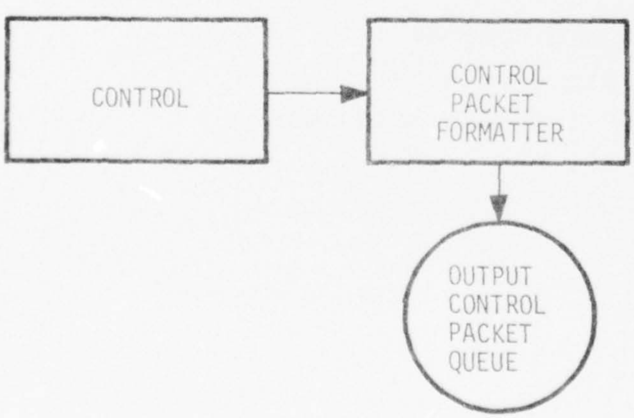


Figure 21. Control Packet Formatter

Receive Table Updater -- The receive table updater (Figure 22) 22) responds to queued requests for changes to the receive table. It updates the receive table to reflect:

- L/S call setup
- L/S call takedown
- L/S call pre-emption

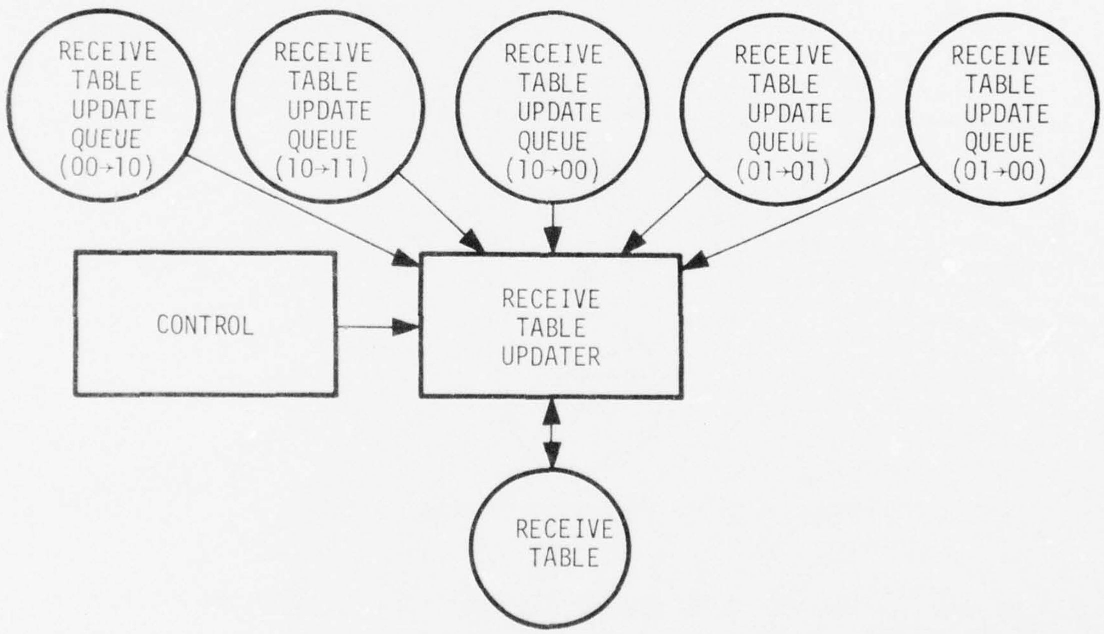


Figure 22. Receive Table Updater

Transmit Table Updater -- The transmit table updater (Figure 23) responds to queued requests for changes to the transmit table. It updates the transmit table to reflect:

- L/S call takedown
- L/S call pre-emption

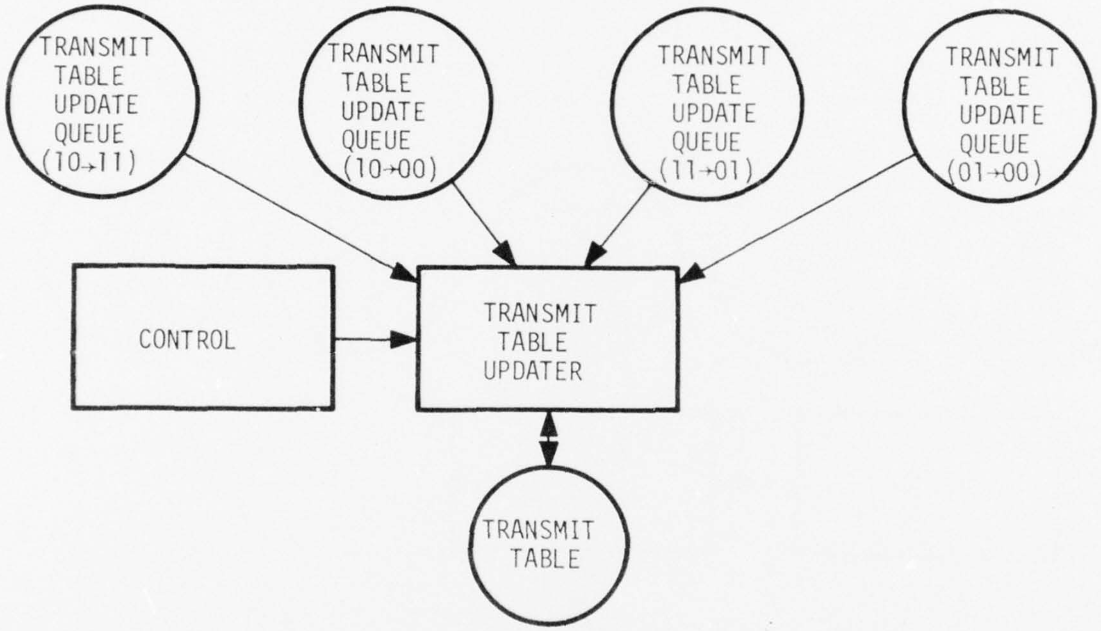


Figure 23. Transmit Table Updater

L/S Router -- The L/S router (Figure 24) responds to queued routing and re-routing requests. It maintains the transmit table (for L/S call setup), the blocked path data base and the routing data base.

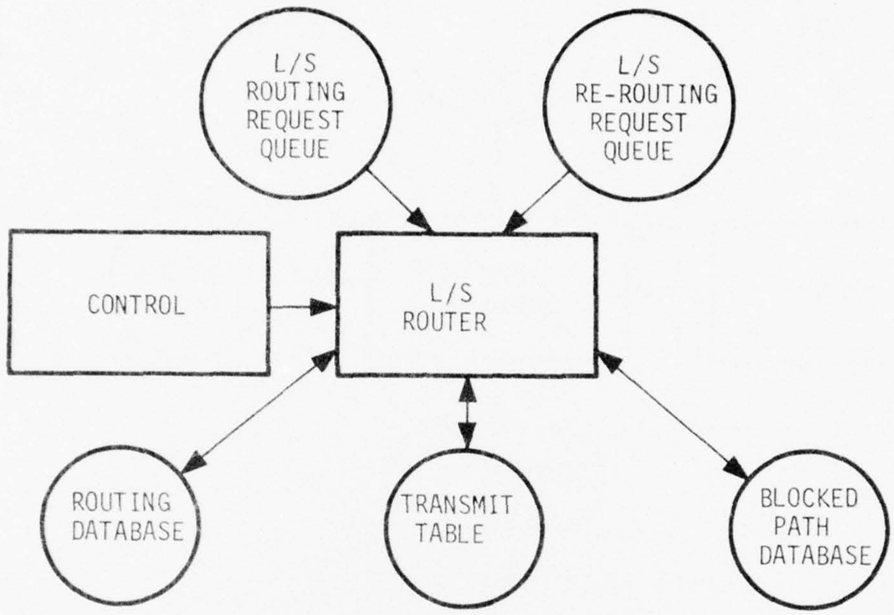


Figure 24. L/S Router

Frame Composer -- The frame composer (Figure 25) creates outgoing frames by combining elements from the following sources:

- Output data packet queue
- Output voice packet queue
- L/S Data
- Output control packet queue
- Sync field data

Frames are formed in accordance with the transmit table and placed on the communications line.

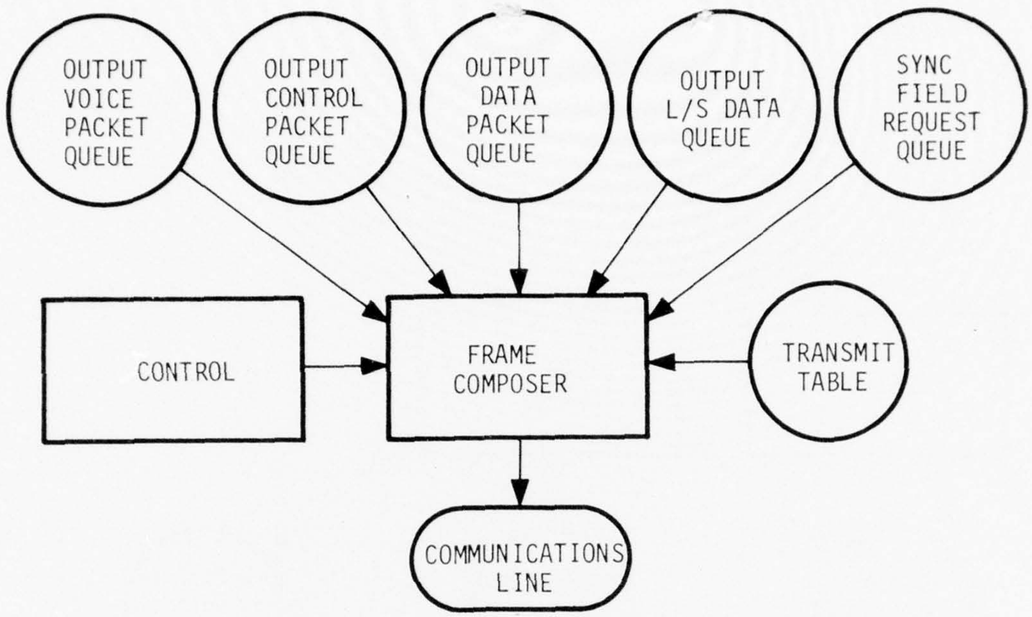


Figure 25. Frame Composer

Process Flow Sequences -- The process flow sequence diagrams depict the sequence of actions performed by a node in response to the receipt of network L/S control packets (types 1, 1a, 2, 3, and 4) and valid synchronization fields. In addition, the sequence leading to the initial transmission of a type 1 control packet is shown. Emphasis is placed on "typical" flow to provide an overall description of node L/S related processing. However, "atypical" flow paths are included for completeness. The notation is informal and is not meant to imply or prohibit concurrent processing.

Initiate Type One (Figure 26) -- In response to a user request for L/S call setup, the node first selects a route. The L/S router examines the routing and blocked path databases and selects an outgoing trunk and time slot appropriate to the call destination and size (bandwidth requirement).

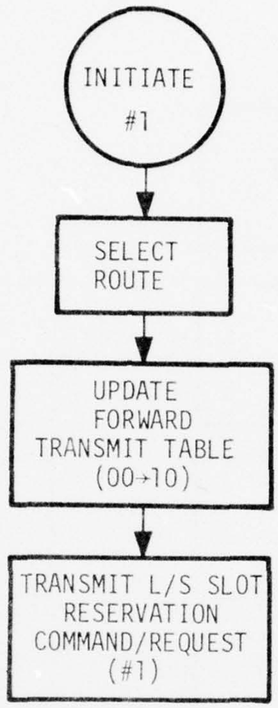


Figure 26. Initiate Type One

The L/S router updates the transmit table corresponding to the selected outgoing trunk to reflect the slot reservation (00 → 10). The control packet formatter creates a type 1 control packet (L/S slot reservation command/request) and places it in the output control packet queue for transmission via the frame composer.

Receive Type One (Figure 27) -- Upon receipt of a type 1 control packet (L/S slot reservation command/request), the control packet processor requests action by the receive table updater and the L/S router. The receive table updater changes the appropriate table entry to indicate the pending transition from P/S data to L/S data (00 → 10). The L/S router searches the routing and blocked path databases to select a return route.

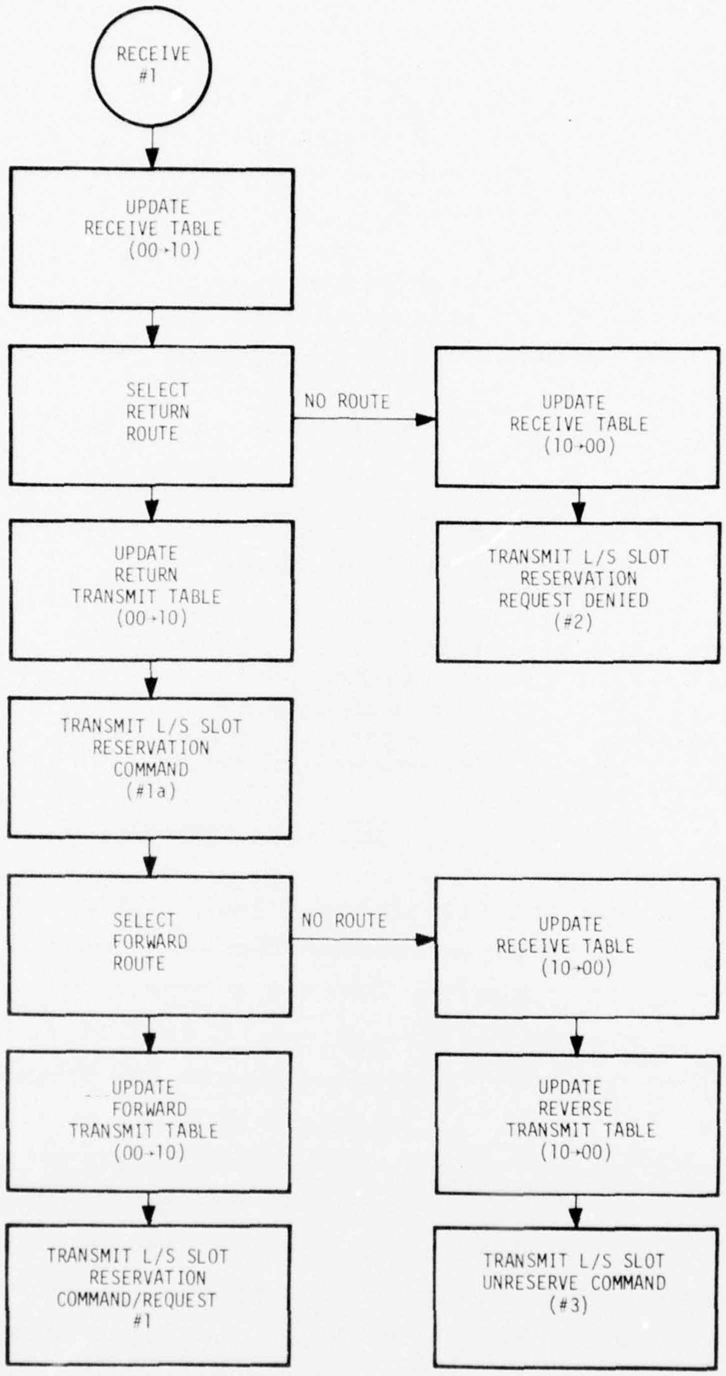


Figure 27. Receive Type One

If no route is found, the receive table updater is requested to restore the receive table to its previous state (10 → 00). The control packet formatter is then charged with creating a "L/S slot reservation request denied" control packet (type 2) for transmission toward the originating node.

If a return route is found, the L/S router updates the transmit table for the return trunk (00 → 10). A "L/S slot reservation command" control packet (type 1a) is then sent along the return path. Next, the L/S router searches the databases to select a forward route.

If no route is found, the receive table updater changes the state of the incoming time slot (10 → 00); the transmit table updater changes the state of the reverse path time slot (10 → 00); and the control packet formatter creates a "L/S slot unreserve command" control packet (type 3) which is sent toward the originating node thereby indicating that an alternate route is required.

If a forward route is found, the L/S router updates the transmit table for the selected trunk and time slot (00 → 10). Then the control packet formatter generates a new type 1 control packet which is sent to the next node.

Receive Type One-A (Figure 28) -- When a node receives a type 1a control packet (L/S slot reservation command), it merely updates the specified receive table entries to reflect the pending transition from P/S data to L/S data. No other action is required.

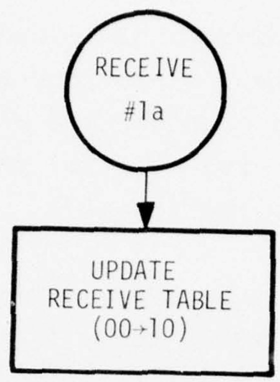


Figure 28. Receive Type One-A

Receive Type Two (Figure 29) -- A type 2 control packet indicates "L/S slot reservation request denied". The node removes the transition pending status of the corresponding forward transmit table entry. Then the L/S router searches for a new forward route.

If no route is found, the path from the originating node's direction must be taken down. The receive table entry is changed from 10 to 00. The transmit table for the reverse direction is also updated (10 → 00), and the previous node is informed of the back routing via a type 3 control packet (L/S slot unreserve command).

If a new forward route is found, the transmit table is updated to reflect the state of the new forward route (00 → 10) and a "L/S slot reservation command/request" control packet (type 1) is sent to the next node.

Receive Type Three (Figure 30) -- A type 3 control packet (L/S slot unreserve command) results in the same process sequence

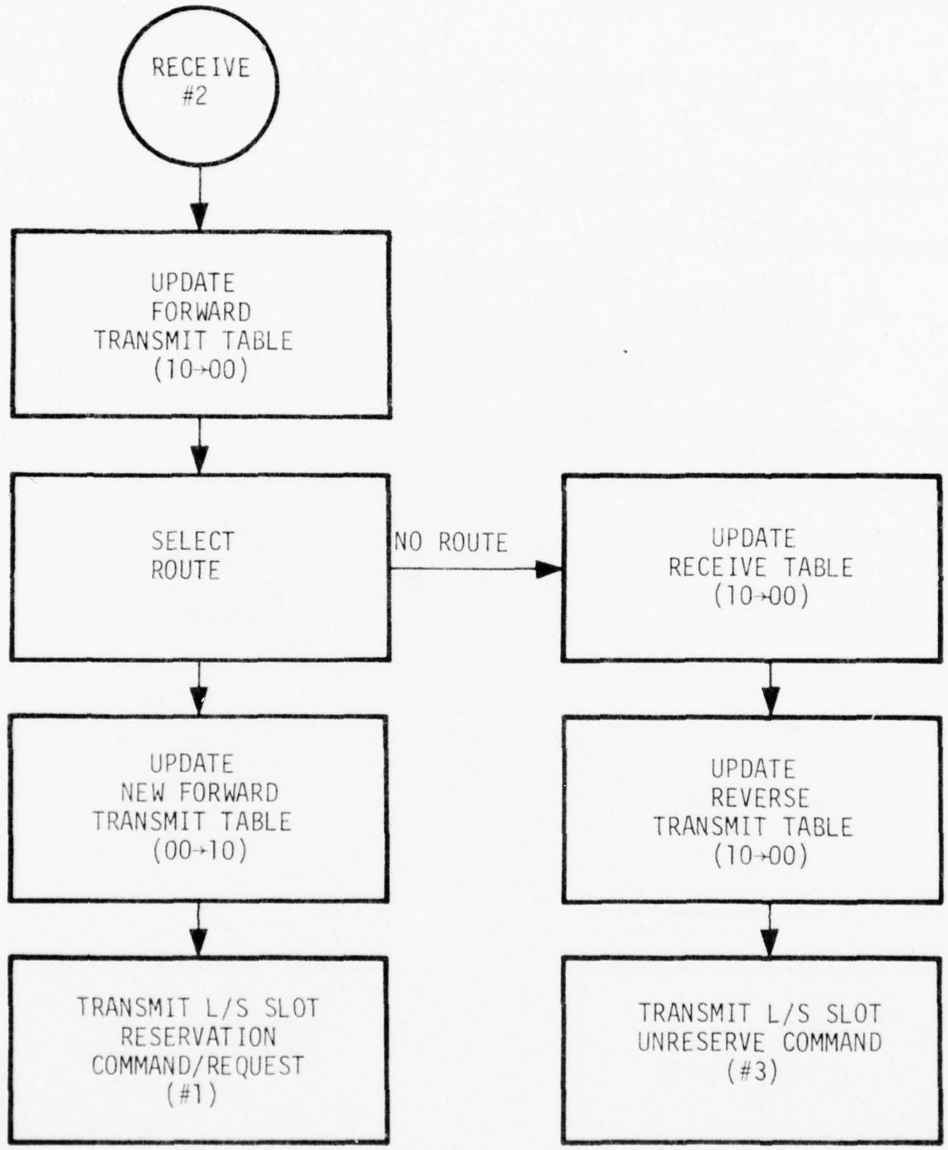


Figure 29. Receive Type Two

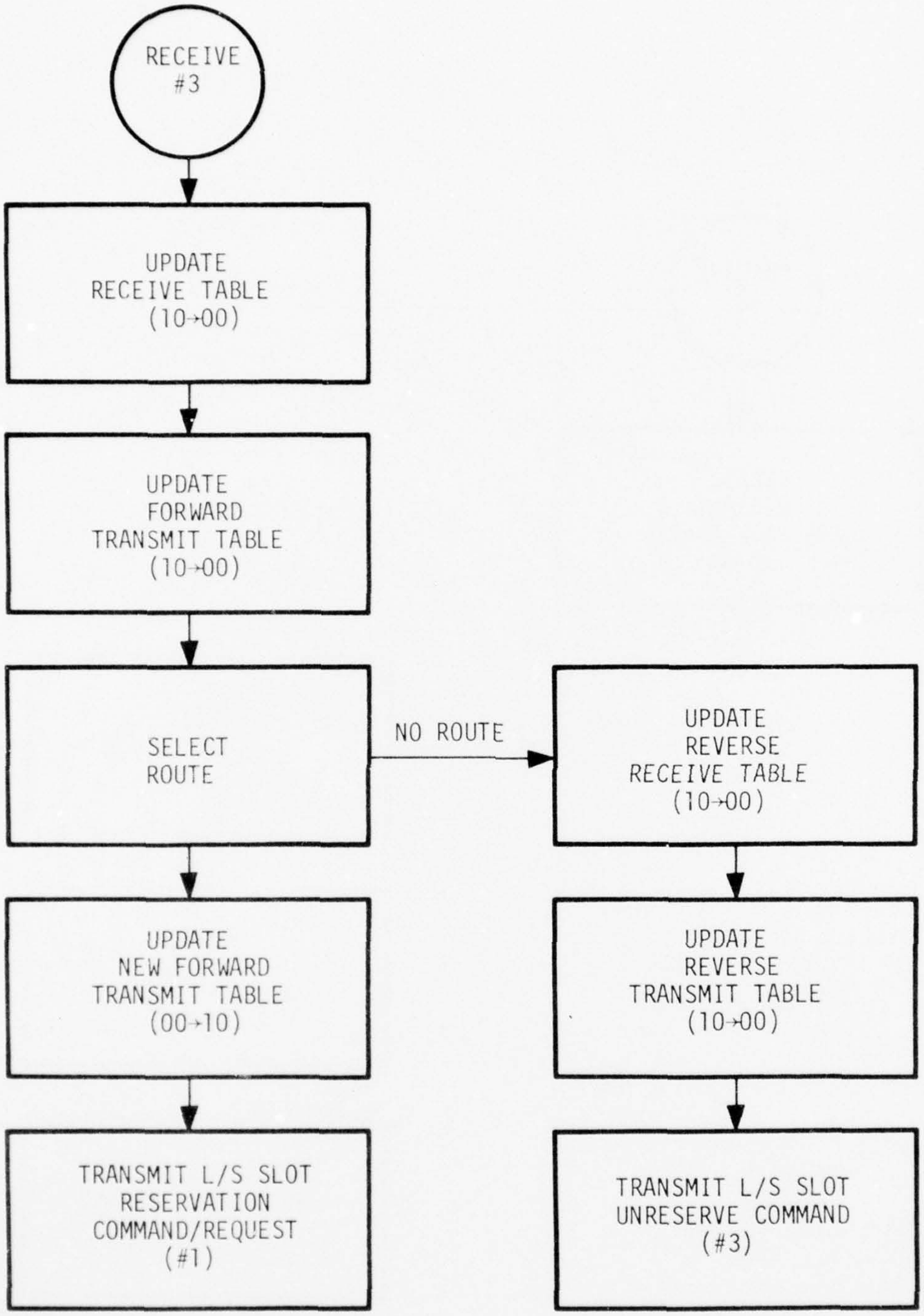


Figure 30. Receive Type Three

as a type 2 control packet except that first the receive table for the reverse path link from the called node direction must be updated to remove the slot reservation (10 → 00).

Receive Type Four (Figure 31) -- Upon receipt of a type 4 control packet ("L/S slot de-reservation command"), the node prepares all four call links for de-allocation by updating the forward and reverse transmit and receive tables to the 01 state. It then propagates the type 4 control packet to the next node along the call path.

Receive Sync Field (Figure 32) -- Valid synchronization fields result in time slot allocation or de-allocation. This is accomplished by appropriate updates to the transit and receive tables in one call direction and the propagation of the valid synchronization field along that same call direction. A call path is entirely "up" (or "down") when the sync field has propagated around the entire call path and returned to the sync originating node.

PACKETIZED-VOICE (P/V) CALL (CLASS IB) SETUP/TAKEDOWN PROCEDURES

Similar to setting up a Class IA voice call, special control packets are used to set up a P/V call. A fixed route is established and is used by all voice packets of the call. Trunk space for the call will be reserved in accordance with the Class I data limit. That is, all P/V calls must be transmitted in the Class I portion of the frame, and should all calls provide samples simultaneously, there will be sufficient bandwidth to transmit all samples. When P/V samples are not present, Class II/III data packets will be used to fill the available trunk capacity.

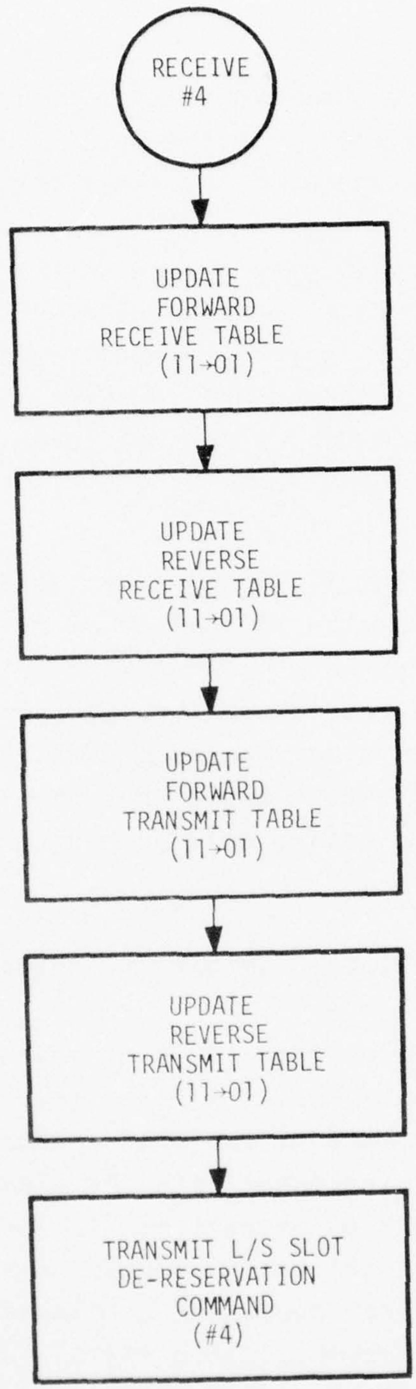


Figure 31. Receive Type Four Control Packet

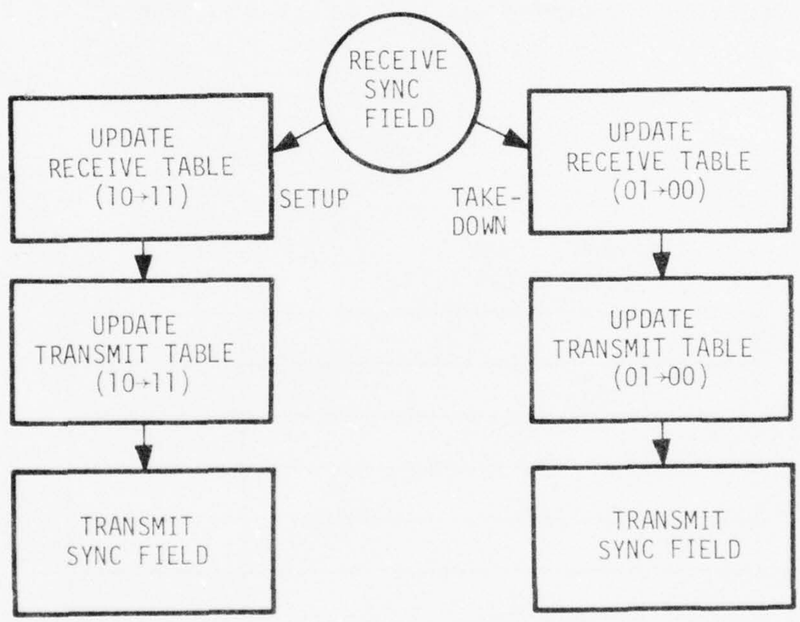


Figure 32. Receive Sync Field

At each node along the path of a P/V call in the network, the voice packets of the call are switched to a fixed output trunk by consulting the P/V call-switching table. At each node the only operation required for setting up or taking down a P/V call is to update its P/V call-switching table. Figure 33 shows a P/V call-switching table at a node. Each row in the table corresponds to a P/V call passing through this node. Each row has two entries: the Directional Call Number and the Output Trunk Number. The rows are created or removed as P/V calls are set up or taken down.

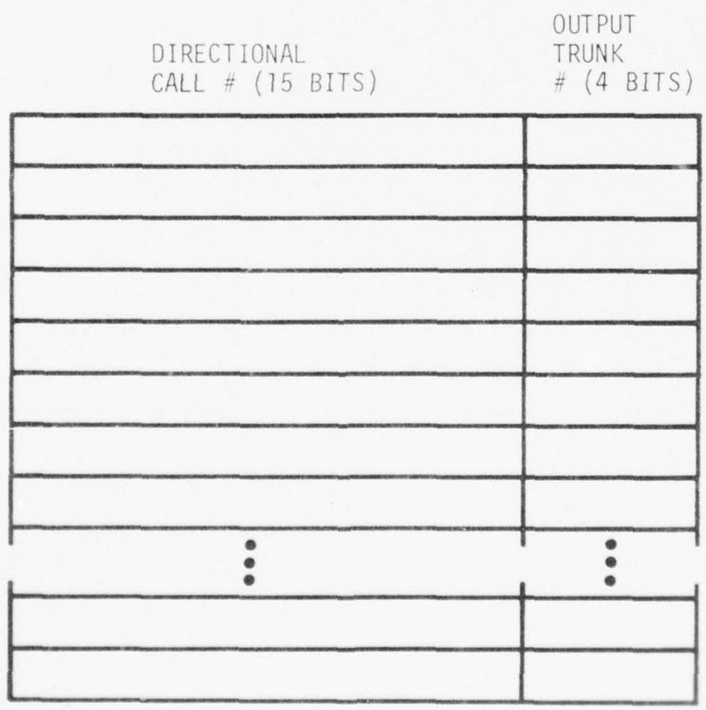


Figure 33. Packetized-Voice Call-Switching Table

AD-A040 636 HONEYWELL INC MINNEAPOLIS MINN SYSTEMS AND RESEARCH --ETC F/G 17/2
APPLICATION OF ASSOCIATIVE PROCESSING TECHNIQUES TO AN INTEGRAT--ETC(U)
JUN 76 H G SCHMITZ, T L SAXTON, C C HUANG DCA100-75-C-0048

UNCLASSIFIED NL

2 OF 2
ADA
040636

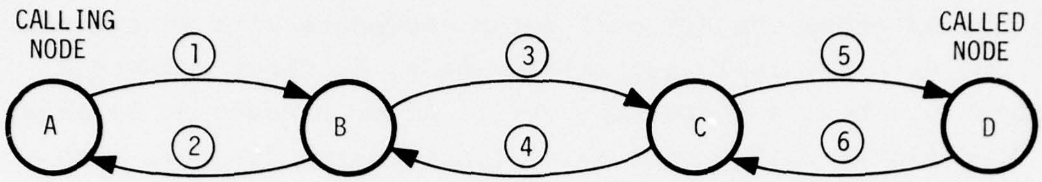
END
DATE
FILMED
7-77

Packetized-Voice Call Setup

Figure 34 shows the P/V call setup procedure with an example. To set up a P/V call, the calling node, A, first selects a route to a next node for the call. After a route is selected, a directional-call-number entry representing the call with direction toward the called node (D) and an accompanying entry with the selected output trunk number are put into a row in the P/V call-switching table at Node A. Node A then sends a Type 10 control packet, "P/V Call Setup Request/Command", to the next node, B. (B is the node at the other end of the selected output trunk from A.)

The operations at a node after it receives a Type 10 control packet are shown in the flowchart of Figure 35. Consider Node B in our example. After receiving the Type 10 control packet from Node A, Node B first checks to see if there is sufficient Class I trunk space available on the reverse-direction line of the selected link between A and B.

If so, then it puts a new row in its P/V call-switching table. The new row has (1) the same 14-bit call number as that put in the P/V call-switching table of Node A; (2) the direction bit flipped to indicate the reverse direction (Node B to Node A); and (3) the trunk number corresponding to the selected link between Node A and Node B. If Node B is the called node, then no more operation is required at B and the call setup is now complete. Because Node B is not the called node (in the example), it will then continue to set up the P/V call by repeating the operations which occurred at Node A. (The operations include selecting an output trunk, creating a new row in the P/V call-switching table, and forwarding a Type 10 control packet to the next node connected by a selected trunk.)



STEP:

0: AT A SELECT A ROUTE AND ENTER A ROW INTO THE P/V CALL SWITCHING TABLE FOR FORWARD-DIRECTION (A TO B) VOICE PACKETS OF THE CALL.

1: SEND A TYPE 10 CONTROL PACKET FROM A TO B.

2: AT B (i) PUT A ROW IN THE P/V CALL SWITCHING TABLE FOR REVERSE-DIRECTION (B → A) VOICE PACKETS OF THE CALL; (ii) SELECT A ROUTE AND ENTER A ROW INTO THE P/V CALL SWITCHING TABLE FOR FORWARD-DIRECTION (B to C) VOICE PACKETS OF THE CALL.

3: SEND TYPE 10 CONTROL PACKET FROM B TO C.

4: SIMILAR TO STEP 2 FOR NODE C.

5: SAME AS STEP 3 .

6: SIMILAR TO STEP 2 - (i).

Figure 34. Packetized-Voice Call Setup Procedure

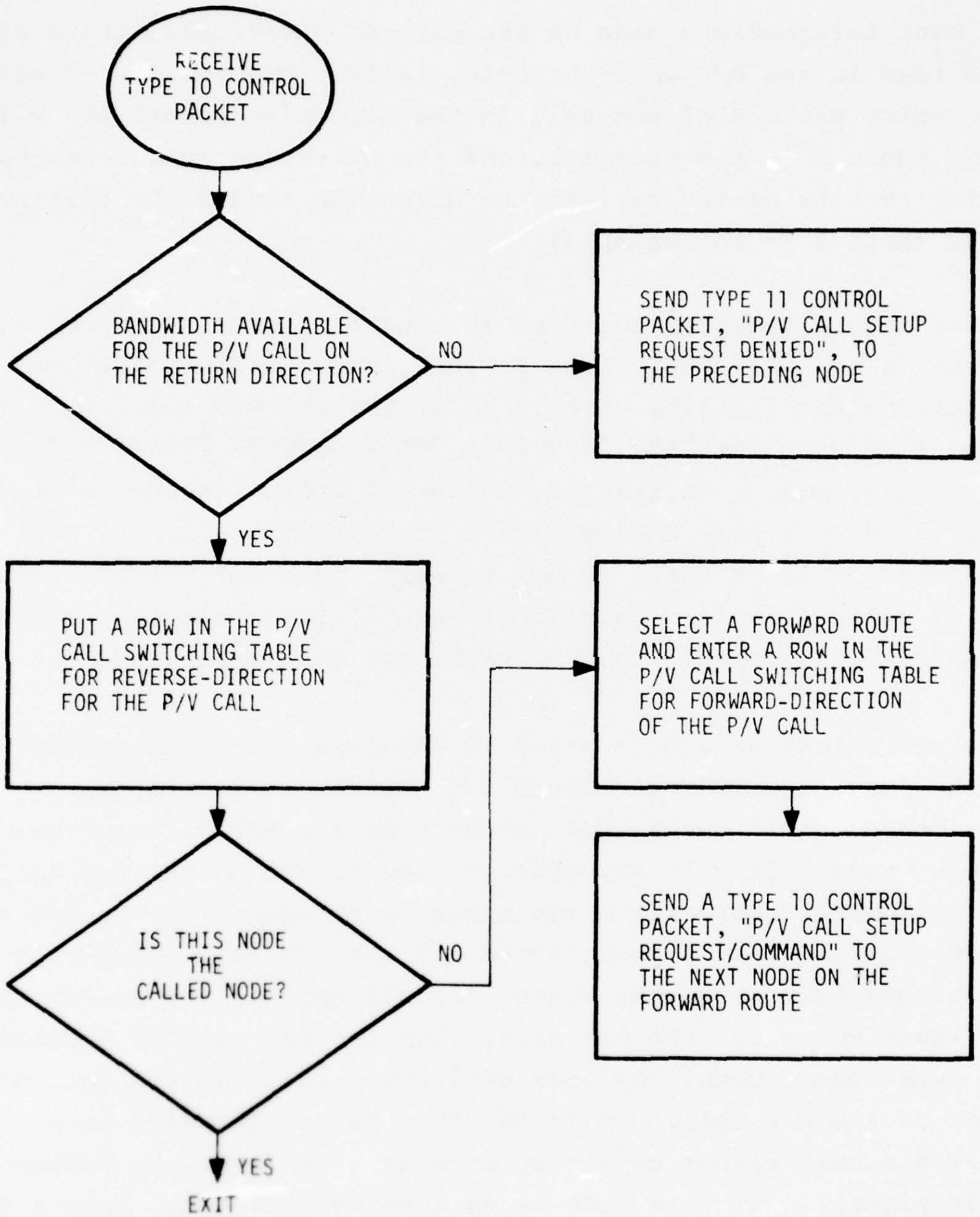


Figure 35. Packetized-Voice Call Setup Flowchart

In each intermediate node on the path of a P/V call, there are two rows in the P/V call-switching table. One row is for switching voice packets of the call in the direction toward the called node (Node D in the example), and the other row is for switching voice packets of the call in the direction toward the calling node (Node A in the example).

After Node B receives the Type 10 control packet from Node A, if there is no sufficient Class I trunk space available on the reverse-direction line (from B to A) for the P/V call, then a Type 11 control packet, "P/V Call Setup Request Denied", will also be sent by Node B to Node A. A Type 11 control packet will also be initiated by a node and sent to a preceding node if the node cannot find trunk space in any forward route for the packetized-voice call. The flow chart in Figure 36 describes the conditions under which a Type 11 control packet will be initiated.

The operations at a node after it receives a Type 11 control Packet are described in Figure 37. First, it deletes a row corresponding to the denied route from its P/V call-switching table. Next, it will try other routes to set up the P/V call. If some other appropriate route has trunk space for the P/V call, then that route will be selected and the P/V call setup will be continued following that route. If all appropriate routes have no trunk space for the P/V call, then the P/V call is blocked at this node. Then, the node will check if it is the calling node of the P/V call. If it is, then the caller will be notified that his call cannot be set up at that time (e.g., by network busy signal). If this node is an intermediate node, then a Type 11 control packet will be generated by the node and sent to the preceding node on the partially set up path of the P/V call.

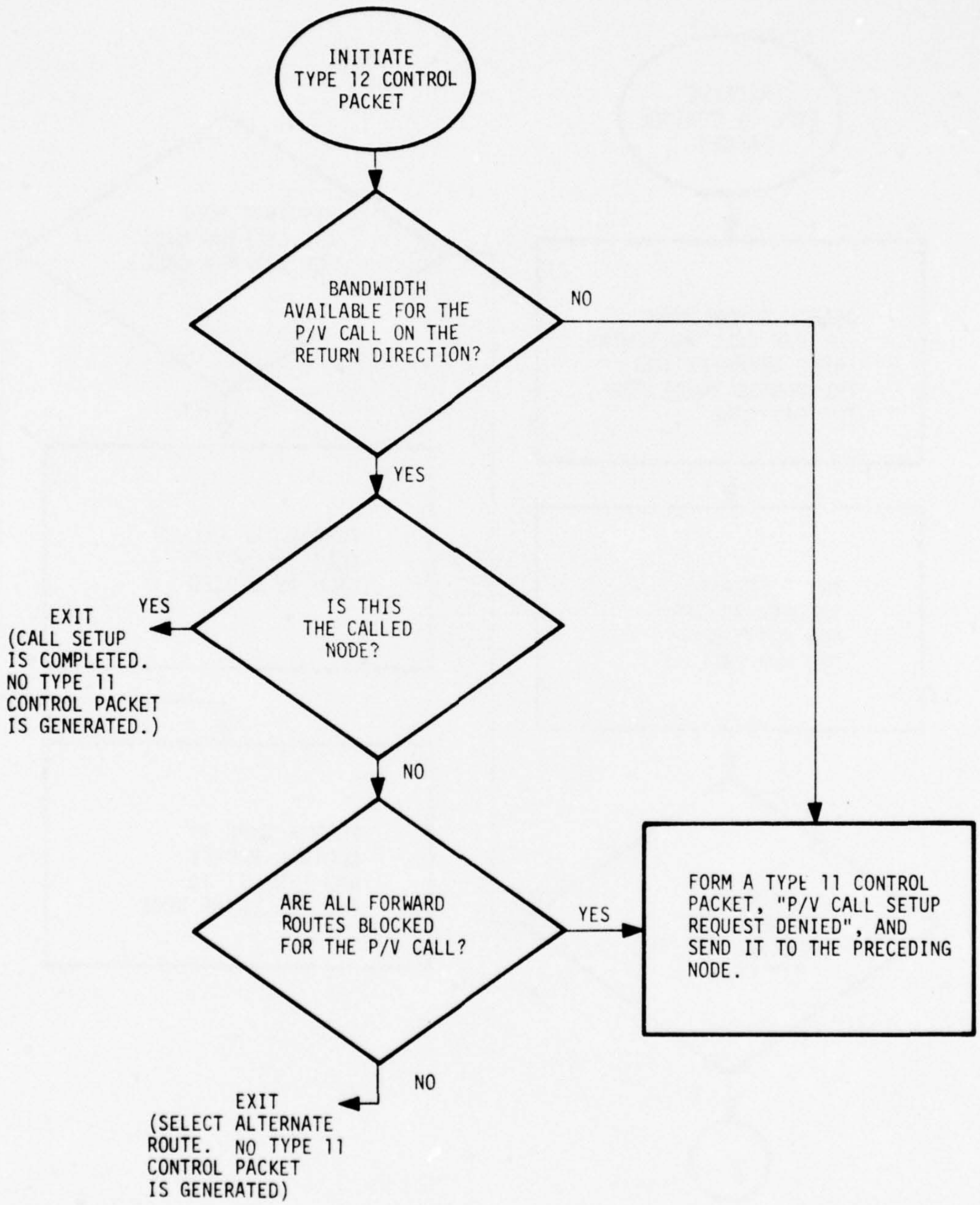


Figure 36. Initiate Type 11 Control Packet Flowchart

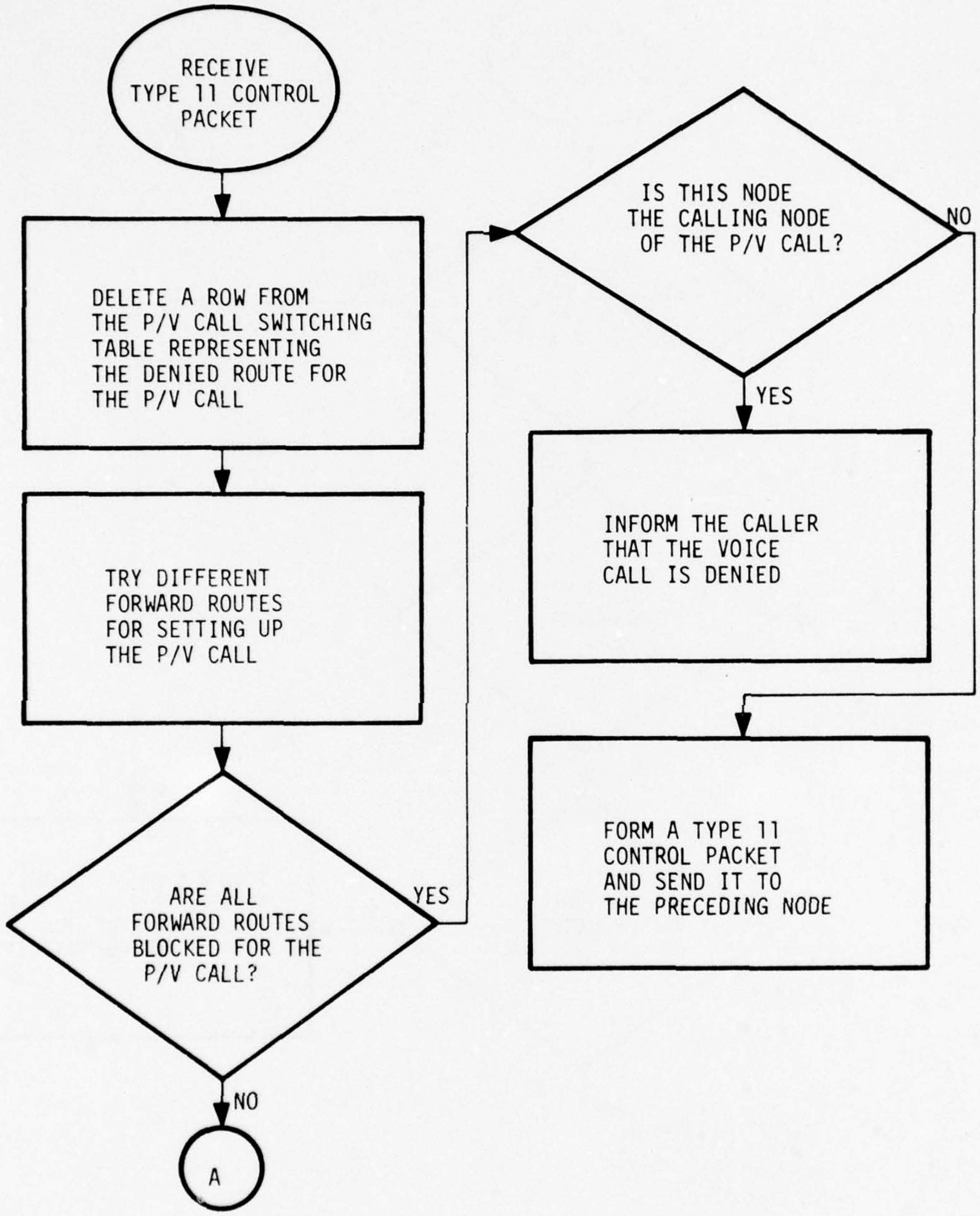


Figure 37. Receive Type 11 Control Packet Flowchart

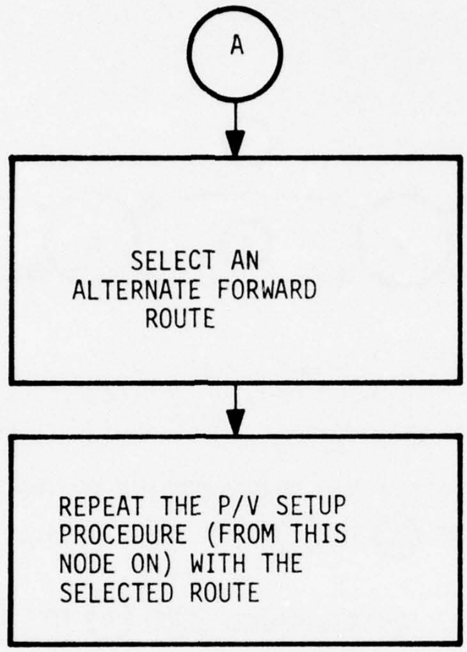
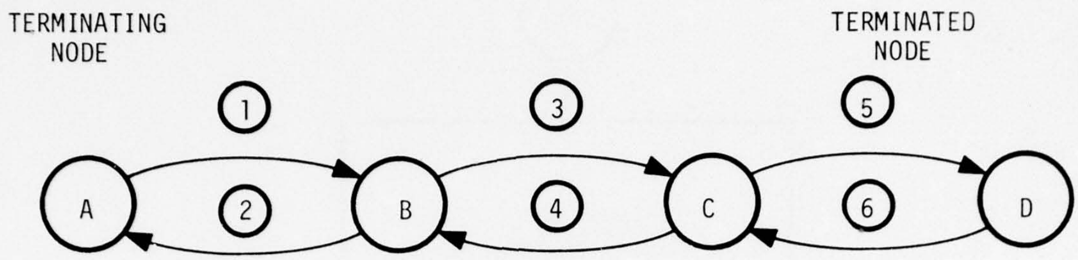


Figure 37. Receive Type 11 Control Packet Flowchart (Concluded)

Packetized Voice (P/V) Call Takedown

A P/V call takedown example is shown in Figure 38. The terminating node, A, first updates its P/V call-switching table by deleting a row representing the call on the link from Node A to Node B. Then Node A generates and sends a Type 12 control packet, "P/V Call-Takedown Command", to the next node on the path of the P/V call being taken down. The call takedown process at Node A is then complete.



STEP:

0: AT A, DELETE A ROW CORRESPONDING TO THE CALL IN THE DIRECTION OF A TO B FROM THE P/V CALL SWITCHING TABLE.

1: SEND TYPE 12 CONTROL PACKET FROM A TO B.

2: AT B, DELETE TWO ROWS FROM THE P/V CALL SWITCHING TABLE, ONE FOR THE P/V CALL IN THE DIRECTION OF B TO A AND THE OTHER FOR THE SAME CALL IN THE DIRECTION OF B TO C.

3: SEND TYPE 12 CONTROL PACKET FROM B TO C.

4: SIMILAR TO STEP 2 FOR NODE C .

5: SAME AS STEP 3.

6: AT D, DELETE A ROW CORRESPONDING TO THE CALL IN THE DIRECTION OF D TO C.

Figure 38. Packetized-Voice Call Takedown Procedure

The operations at a node after it receives a Type 12 control packet are shown in the flow chart of Figure 39. Consider Node B in this example. After receiving the Type 12 control packet from Node A, Node B updates its P/V call-switching table by deleting a row representing the call on the link from Node B to Node A. If Node B were the terminated node, then the P/V call-takedown process would be complete. In this example, Node B is an intermediate node. Therefore, Node B will then continue the P/V call-takedown process by repeating the operations executed by Node A, which include deleting a row (link from B to C) from the P/V call-switching table (at Node B) and generating and sending a Type 12 control packet to the next node (C) on the path of the P/V call.

PACKET-SWITCHED (P/S) DATA TRANSMISSION PROCEDURES

Packet Transmission Procedures Between Source Node and Destination Node

Source to Destination Node Protocol for Class II/III Data Packets -- The proper source node to destination node protocol is presented in the following subsections.

Connection Setup Between Source Node and Destination Node -- To transmit Class II or Class III data from a user terminal or computer process port (called a source port) to another user terminal or computer process port (called a destination port) requires the establishment of a one-direction logical connection between these two ports. The source port sends a Type 5 Control Packet, "P/S data transmission connection request", to the desintation node to inquire whether the destination port is "on". The destination node, in return, sends a Type 6 Control Packet, "P/S data transmission connection answer", to the source node. If the

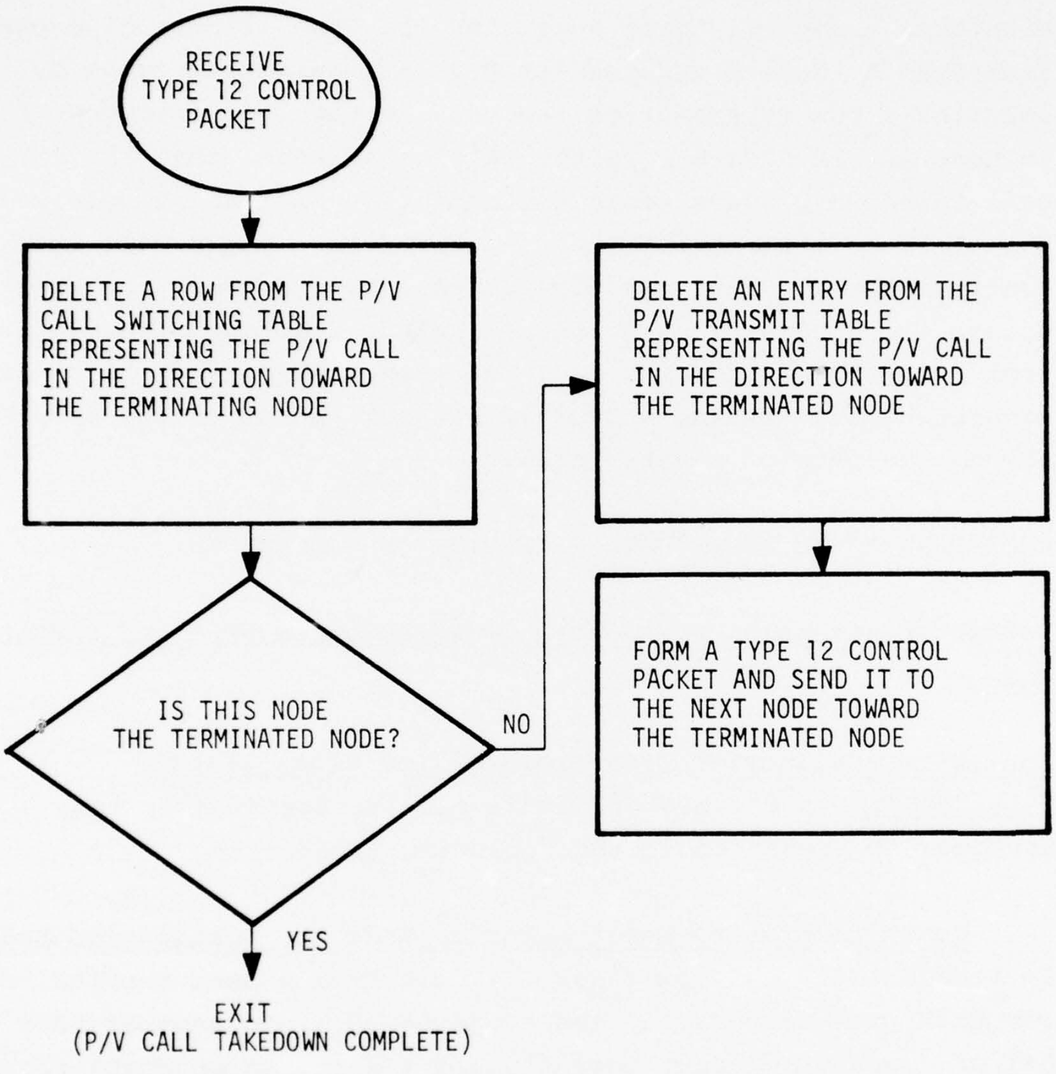


Figure 39. Receive Type 12 Control Packet Flowchart

destination port is "on", then the returned Type 6 Control Packet is affirmative; and the connection is established when the source node receives the Type 6 Control Packet. The P/S transmission connection setup procedure is shown in Figure 40. The operations at a node after it receives a Type 5 control packet are shown in Figure 41. The operations at a node after it receives a Type 6 control packet are shown in Figure 42.

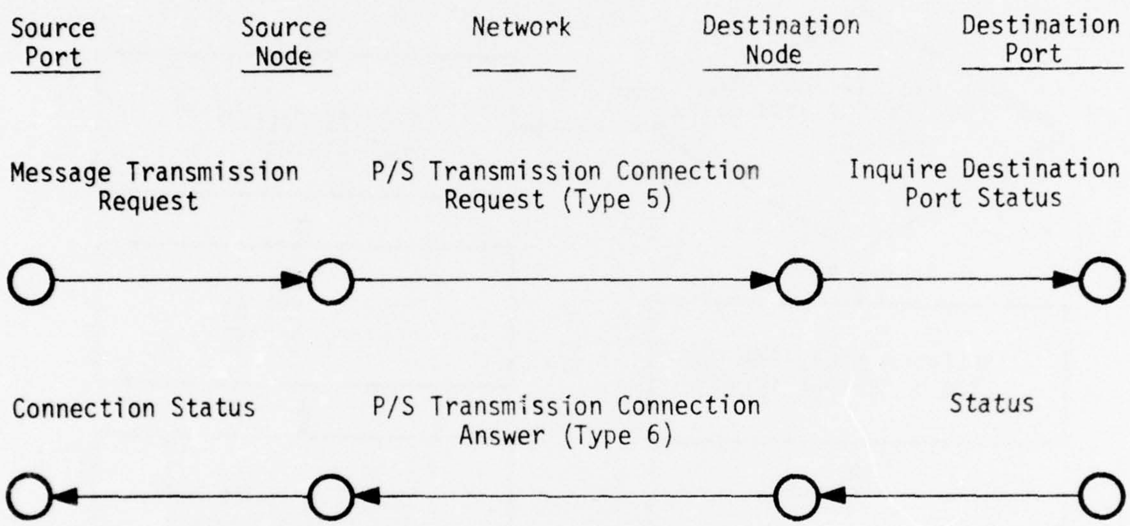


Figure 40. P/S Transmission Connection Setup

After the one-direction logical connection is made, the source node will attempt to allocate buffer space for one message for the source port. If the buffer space can be allocated, then the source port will start to load the buffer with a message. If the buffer space cannot be allocated for a predetermined amount of time, then the logical connection will be cleared (timed out). If the buffer space is successfully allocated at the source node for this connection, then actual transmission of messages over the network will be initiated. Depending on the length of each message, two different procedures are used for transmitting the messages (which are packetized) from the source port to the destination port.

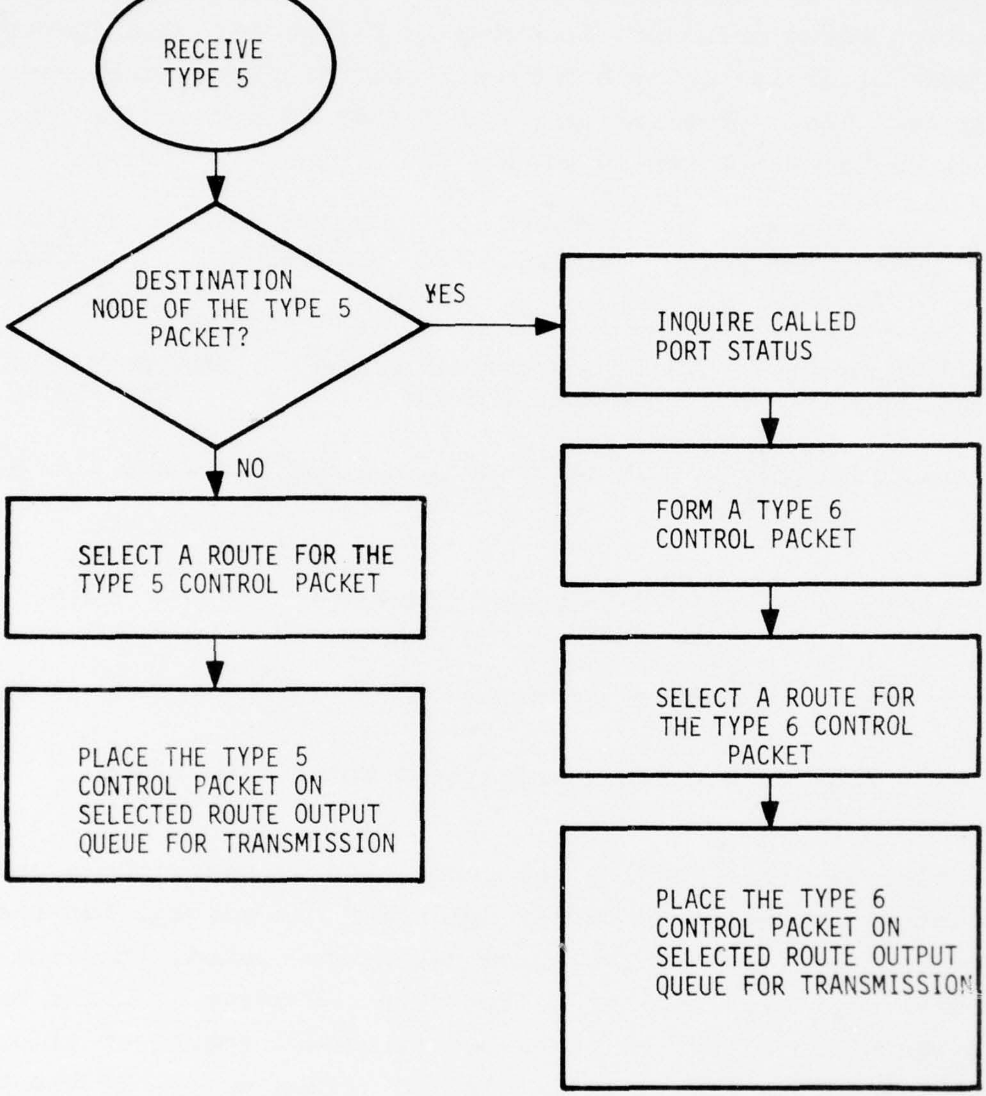
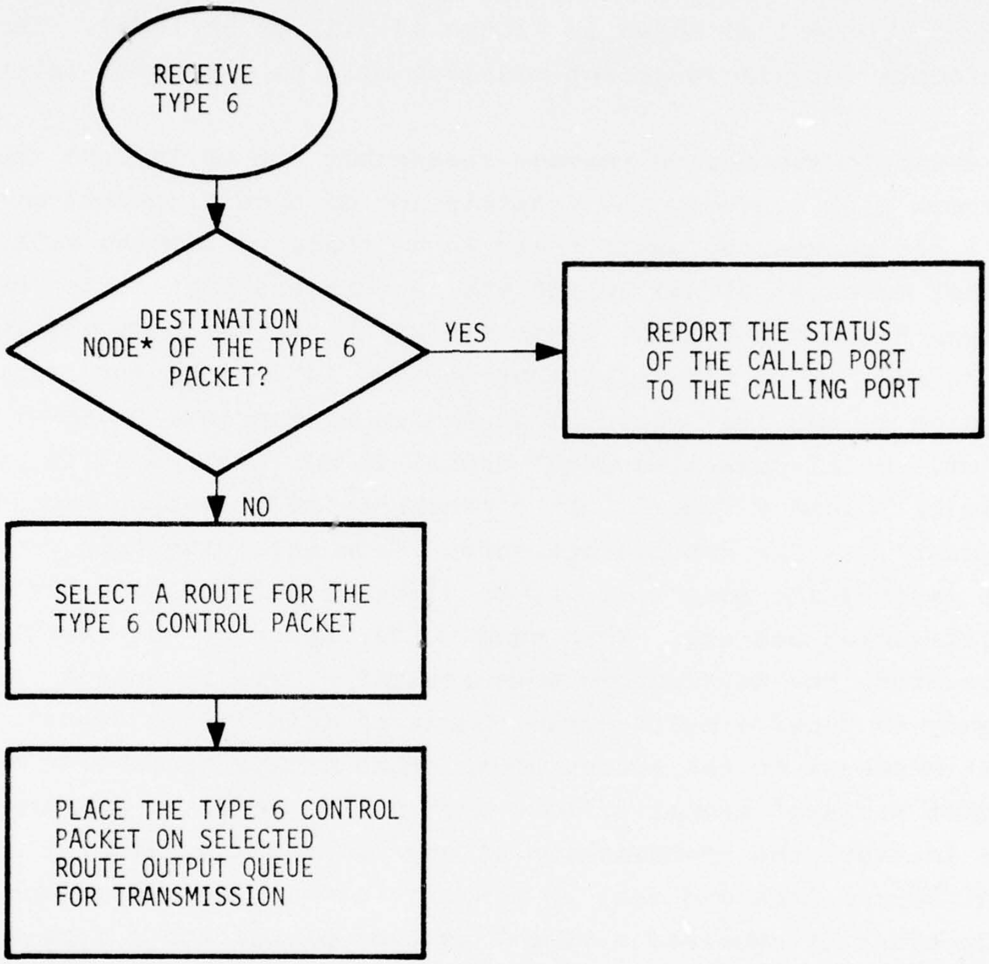


Figure 41. Receive Type 5 Control Packet



*THIS IS THE SOURCE NODE THAT INITIATES THE SETUP OF THE P/S TRANSMISSION CONNECTION CONCERNED.

Figure 42. Receive Type 6 Control Packet

Multi-Packet Message Transmission Procedure -- For long messages, which cannot be contained in one packet, a multi-packet protocol as shown in Figure 43 will be employed. The protocols for single-packet messages will be discussed later.

To avoid lockup during message reassembly and to improve the network flow control, the transmission of a multi-packet message will not be started until there is a "ready to receive multi-packet message" signal (which was sent by the destination node via a Type-8 Control Packet) available at the source node waiting to be "used". When a multi-packet message is pending for transmission to the destination node and an appropriate "ready to receive multi-packet message" signal is not available, the source node will send a Type-7 Control Packet, "multi-packet transmission request", to the destination node. Upon receiving this request, the destination node will try to allocate buffer space for one multi-packet message. When this buffer space is successfully allocated, the destination node returns a Type-8 Control Packet, "Ready to receive multi-packet message" (similar to Request for Next Message) to the source node. Each "ready to receive multi-packet message" signal (from a destination node) at a source node can initiate the transmission of one multi-packet message between that source node and that destination node. The operations at a node after it receives a Type-7 control packet and a Type-8 control packet are shown in Figures 44 and 45, respectively.

Each multi-packet message is sent as several packets from a source node to a destination node. After the destination node has received all packets of the message, it reassembles them into a message and outputs to the destination port. Upon completion of the output process, it sends a Type-8 Control Packet, "ready to receive a multi-packet message", to the source node, since it has buffer space available to accept a new multi-packet message.

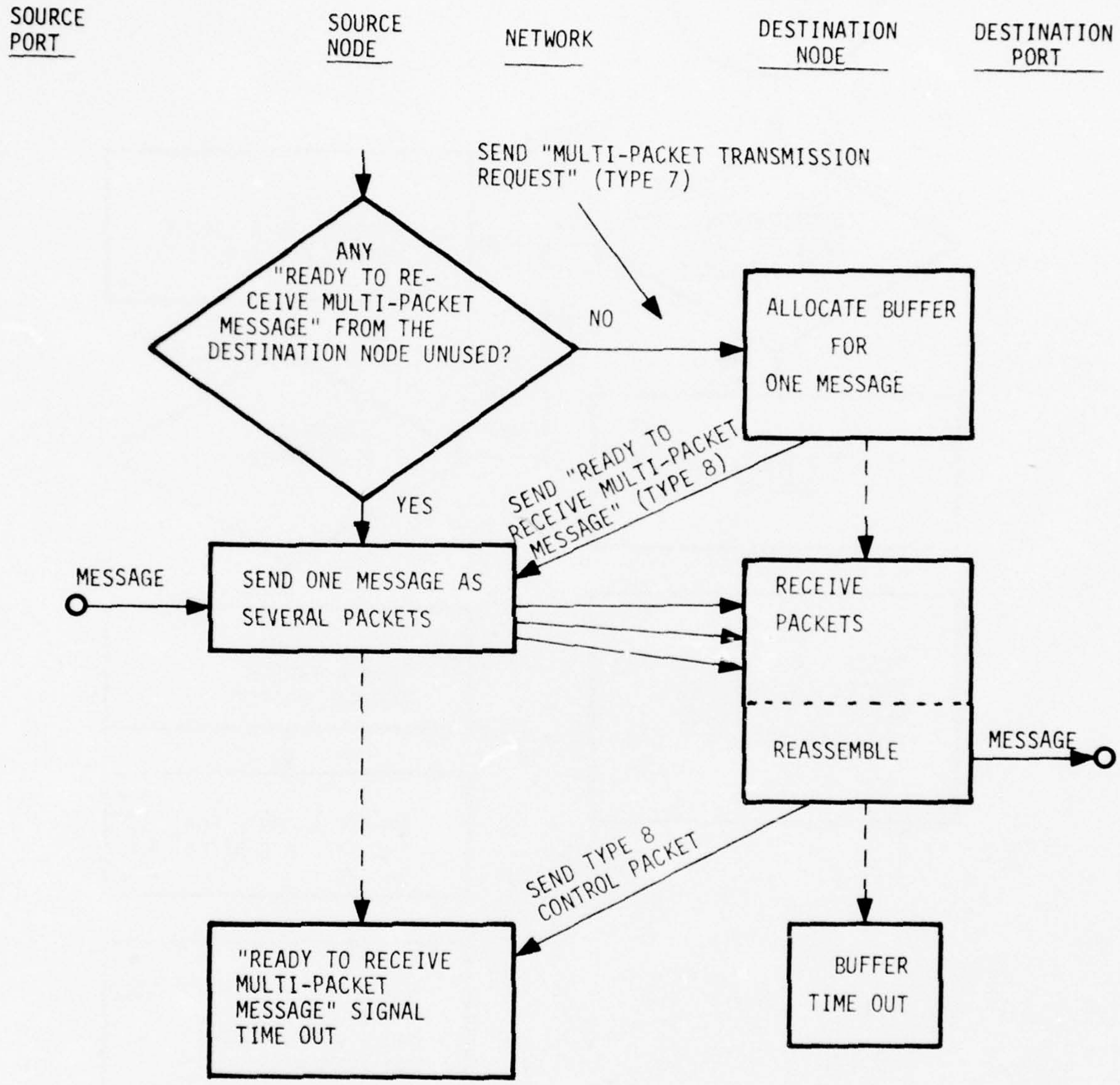


Figure 43. Multi-Packet Message Transmission Procedure

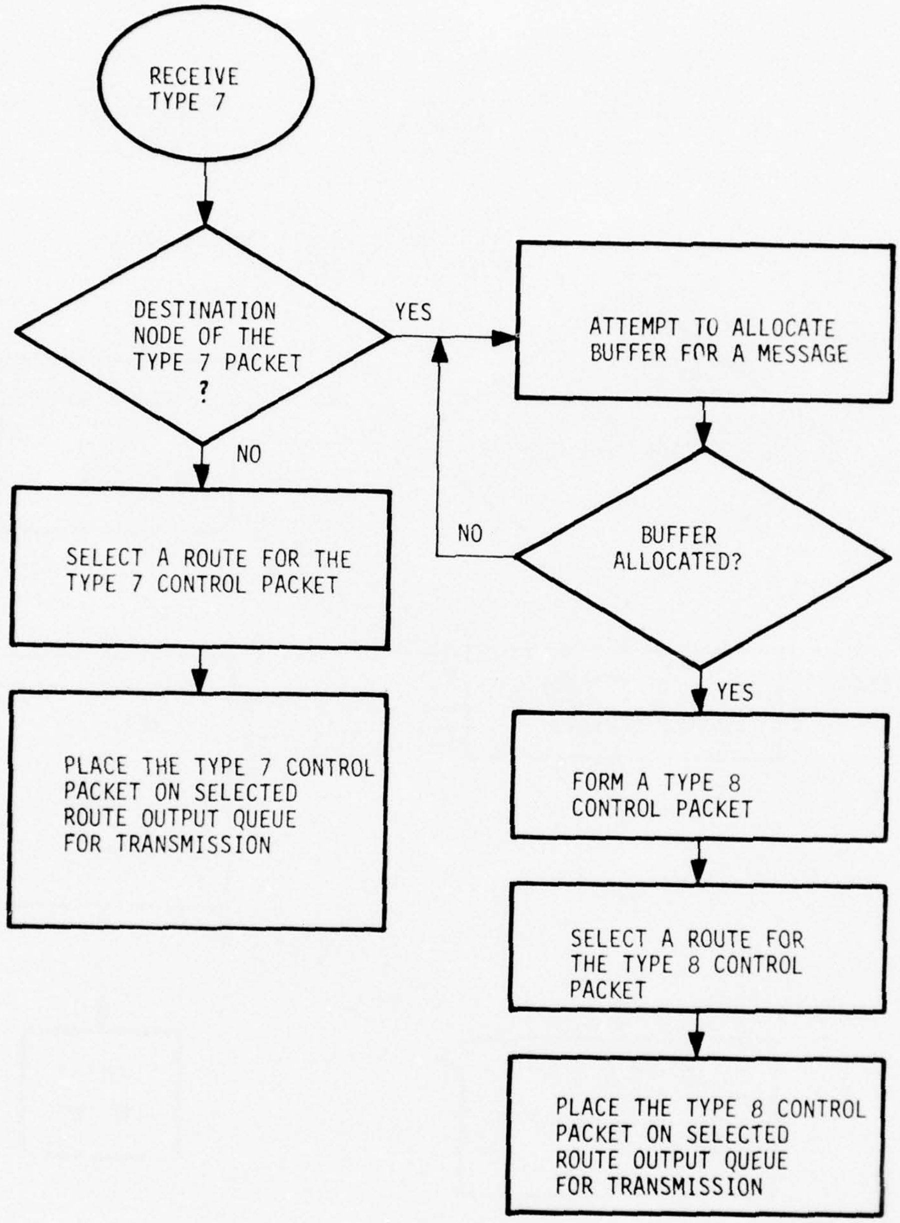


Figure 44. Nodal Processing after Receiving Type 7 Control Packet

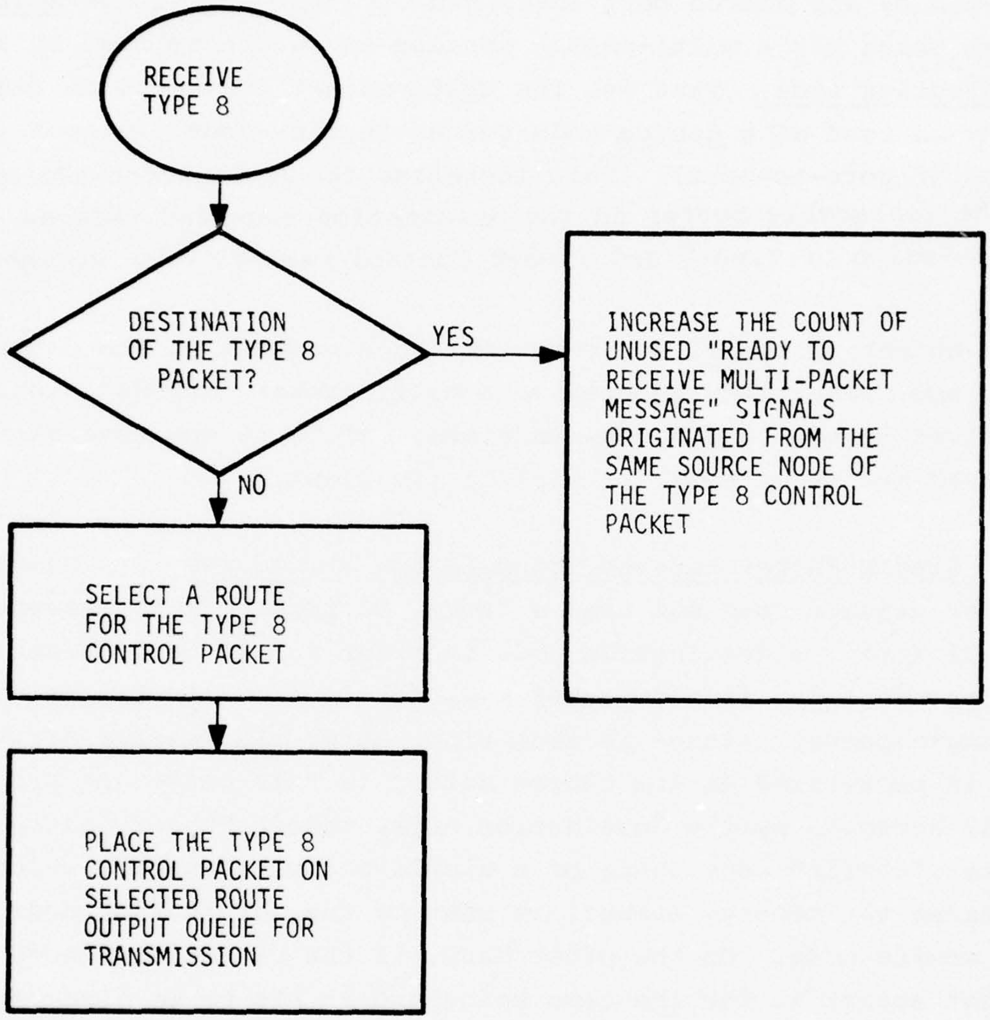


Figure 45. Nodal Processing after Receiving Type 8 Control Packet

Note that the "ready to receive a multi-packet message" signal can be used by any source port (belonging to that particular source node) which has a multi-packet message to be transmitted to the destination node. That is, the multi-packet transmission request/ready is used as a source-node-to-destination-node protocol (instead of port-to-port). This increases the utilization efficiency of the reassembly buffer in the destination node and reduces the transmission of Type-7 and Type-8 Control Packets over the network.

Each buffer allocated for reassembling a message in the destination node and each "ready to receive a multi-packet message" signal received in the source node is timed. If it is not used within a fixed amount of time, it will be timed out.

Single-Packet Message Transmission Procedures -- A single-packet message does not need a "ready to receive next message" signal from its destination node in order for it to be sent. The protocol for single-packet messages is shown in Figure 46. A single-packet message is sent right after the message arrives and is packetized at the source node. If this packet is successfully accepted by the destination node, then a Type-9 Control Packet,"confirm acceptance of a single-packet message", which contains the message number, is sent by the destination node to the source node. On the other hand, if the destination node cannot accept it for the time being and it has to be discarded, then no acknowledgement will be returned by the destination node to its preceding node, which had attempted to send the packet (refer to "Node-to-Node packet protocol on page 104). The packet will be resent after a timeout. When the packet is successfully accepted by its destination node, then a Type-9 Control Packet, "confirm acceptance of a single-packet message", will be sent by the destination node to the source node. The operations at a node

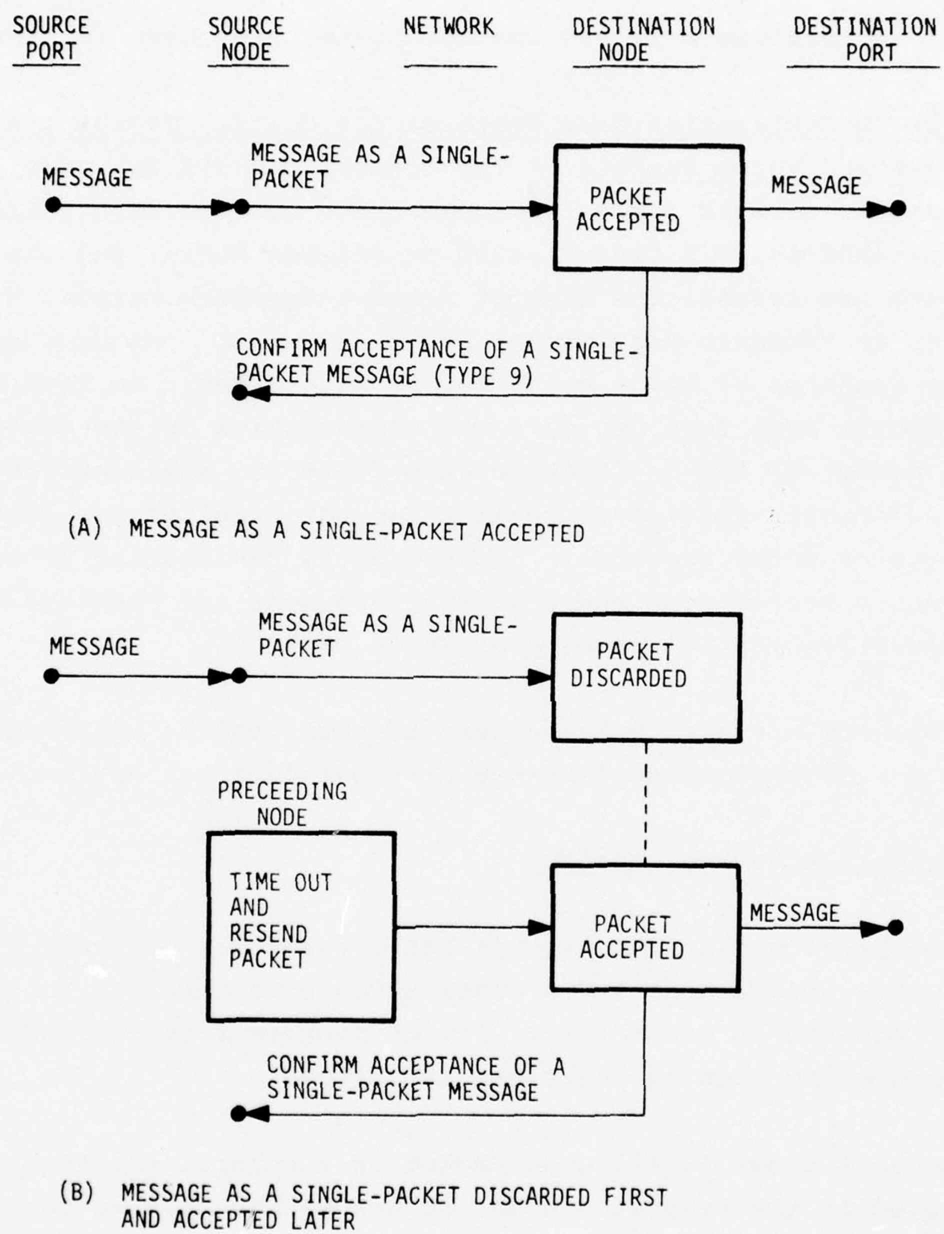


Figure 46. Single-Packet Message Transmission Procedures

after it receives a Type-9 control packet are shown in Figure 47.

Source to Destination Node Protocol for Control Packets, Ack Packets and Voice Packets -- The control packets are generated within the network for network operation control (e.g., L/S setup/takedown, P/S transmission procedures,etc.); and the voice packets are transmitted without acknowledgements between nodes (refer to "Node-to-Node packet protocol" below), so that the fixed-delay features of voice calls may be maintained. In both cases the source node does not need the confirmation of the acceptance of a packet by the destination node (that is, the source-node-to-destination-node accountability is not required for control packets or voice packets). Therefore, no handshaking procedures between a source node and a destination node are required for transmitting control packets or voice packets.

Ack packets are used only between adjacent nodes, and hence the source-node-to-destination-node protocol does not apply.

Node-to-Node Packet Protocol

In the previous section, the packet transmission procedures between a source node and a destination node were described. The procedure to control the flow of packets from one node to a neighboring node is described as follows.

Whenever a Class II/III data packet or a control packet is accepted by the receiving node, an Ack packet will be returned to the sending node as acknowledgement. (However, there is no acknowledgement for an Ack packet or a Voice packet.) The sending node is allowed to send Data or Control packets without having received acknowledgements for earlier ones. A one-packet-at-a-

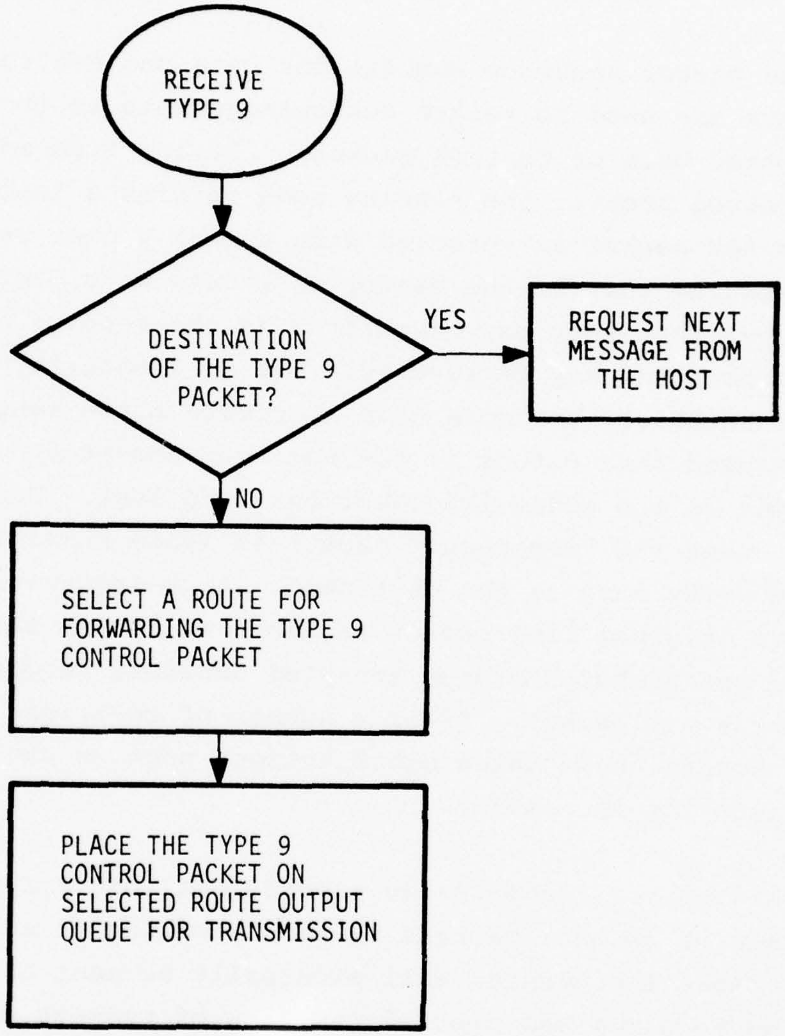


Figure 47. Nodal Processing after Receiving a Type 9 Control Packet

time method will not be used because it is inefficient on long lines.

The packet sequence numbers for Data and Control packets over a link are used to relate acknowledgements to the originally transmitted Data or Control packets. Having transmitted a Data or a Control packet, the sending node retains a "security" copy. If an Ack packet is received with a link packet sequence number matching that in the header of a Data or a Control packet security copy, that copy may be deleted in the sending node and the space it occupied may be re-used. But if a security copy remains unacknowledged for more than a predetermined length of time, it is assumed that either it has not been accepted by the receiving node or the acknowledgement has been lost. Under these circumstances the "timed-out" packet is again transmitted, but its security copy is not destroyed. A re-transmitted packet carries its original link packet sequence number so that the recipient, if the packet had been accepted earlier, recognizes that it is not a new packet. After a number of re-tries, without success, a sending node takes other action, such as choosing an alternate route for the packet.

This packet transmission procedure allows a receiver to ignore data or control packets by simply making no acknowledgement because the packets will eventually be sent again. In this way, the receiver can control the flow of packets on an incoming link according to its operational needs. If the receiver is in its deaf state when a packet starts to arrive, the receiver will never be made aware of its appearance (and hence no acknowledgement will be sent for it). If the packet is received but an error is detected by the hardware, or if the packet is successfully received but cannot be accepted by the node control programs for internal reasons, then no acknowledgement will be generated either.

There is no error checking for Ack packets and voice packets. There is no acknowledgement generated or sent after an Ack packet or a voice packet is received by a node. The sending node of the Ack packet or the voice packet does not retain a security copy for the packet, and these packets are never re-transmitted.

Packet Transmission Procedure Summary

Packet Type	SN-to-DN Connection Setup and Message Acknowledgement	Node-to-Node Packet Acknowledgement
Data Packet (Class II/III)	yes	yes
Control Packet	no	yes
Ack Packet	no	no
Voice Packet (Class IB)	no	no

L/S AND P/S ROUTING CONSIDERATIONS

The complete problems of L/S and P/S routing are outside the scope of the current study program. However, some analysis of these problems is necessary for accurate definition of the boundary between the routing problems and the node and network studies. This section presents the result of that analysis in the areas of:

* Routing Data Base
* Line Switched Routing
* Packet Routing

Our intent is to show both limits and assumptions which may have influenced the analysis of node and network requirements.

Routing Data Base

To facilitate routing of packets and line switched calls, each node must be supplied with a data base which describes:

* Node and network structure
* Dynamic node and network status

The information provided must be sufficient to support the routing algorithms which the node will perform. The algorithms will not be presented here. Only the type of data and the method of its collection will be considered.

Node and Network Structure--At each node, a local data base describes the node and its lines and trunks. This data base includes:

* Number of lines and trunks
* Length (delay) and capacity of each line or trunk
* Queue and buffer capacities
* Deterministic factors

The deterministic factors play a major role in the routing process. They include variables, thresholds, and other parameters which are consulted during execution of the routing

algorithms. They impose routing guidelines on the network, thereby preventing endless circulation of packets and removing the incoming link from consideration during route selection. These deterministic factors may also be used to control the L/S to P/S trunk occupancy ratio or to temporarily remove trunks or nodes from service.

Dynamic Node and Network Status--Some measure of network activity and congestion is required if the routing algorithms are to be successful. The local data base provides this measure by including:

- Queue and buffer occupancy
- Precedence of calls and packets
- Age of queued packets
- Predicted transmission times

The topic of predicted transmission time deserves further discussion. Of particular interest is the computation, communication, and use of this parameter. Figure 48 shows a hypothetical network structure which will aid in understanding these three aspects of predicted transmission time. The examples center on node six, specifically showing routing from node six to nine.

Figure 49 depicts that portion of the local data base known as the predicted transmission time tables. One table lists the delays caused by transmission queueing (described in the packet routing discussion). The remaining tables list predictions of the transmission time to each destination node. For every combination of destination and precedence, the tables list the predicted transmission time via each outgoing trunk.

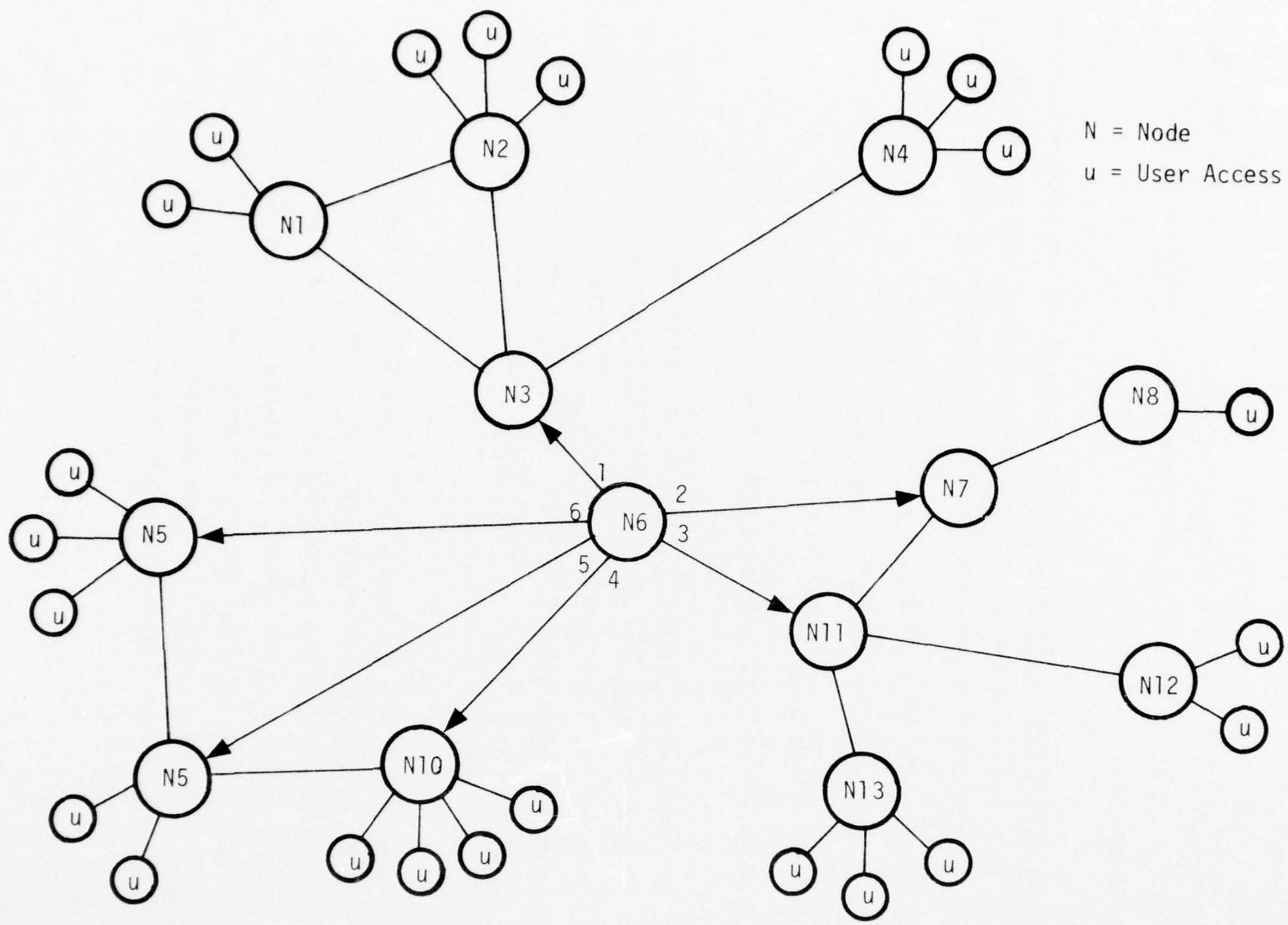


Figure 48. Network Structure

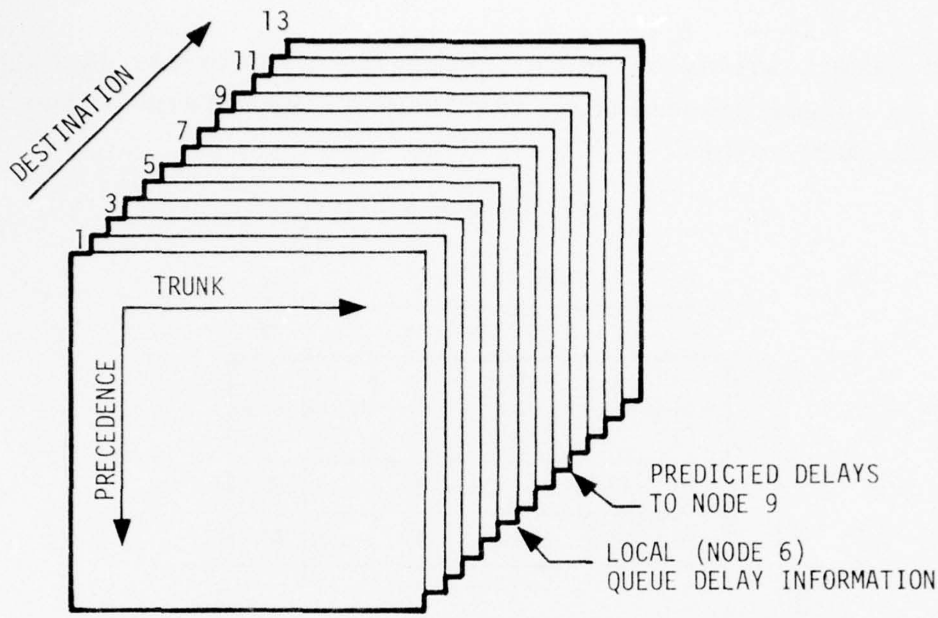


Figure 49. Predicted Transmission Time Tables

New predictions are periodically broadcast to adjacent Nodes (3,5, 7, 9,10 and 11 in the example) to ensure that the total distributed data base accurately describes the current state of the network. When transmission time predictions are received, they are stored in the appropriate tables for use during the routing process. A table which might exist at Node 6 is shown in Figure 50. The column at the right lists predictions which will be broadcast.

The deterministic factors can be used to guide the process of making transmission time predictions. For example, they might eliminate Trunks 1, 2, and 3 from consideration during routing to Node 9. Then predicted transmission times from Node 6 to Node 9 could be based on the predicted values received along Trunks 4, 5, and 6.

While efficient processing of large data tables poses little problem

for parallel and associative processing techniques, it might be wise to reduce transmission requirements by interpolation of intermediate values.

TABLE FOR NODE NINE:

PRECEDENCE	TRUNK 1	2	3	4	5	6	PRE.
1	67	71	70	32	31	37	33
2	73	78	77	35	34	41	36
3	80	85	84	38	37	44	39
4	86	92	90	41	40	48	42
19	181	194	191	87	86	99	89
20	188	201	198	90	89	103	92
21	195	208	205	93	92	107	95
22	202	215	212	96	95	111	99

UNITLESS HYPOTHETICAL VALUES

Figure 50. Predicted Transmission Time Table

Line Switched Routing

The objective of line-switched routing is the rapid establishment of a bi-diretional, port-to-port connection which exhibits a fixed, minimal transmission delay. The procedures for normal call setup and takedown have been presented previously. This discussion centers on route selection. The topics addressed are:

- Route selection criteria
- Alternate route selection
- Back routing

Route Selection Criteria--Line-switched route selection is based on the following criteria:

- Precedence of existing L/S and P/S
- Line or trunk occupancy vs. capacity
- Line or trunk length (delay)
- Deterministic factors

Predicted transmission time can be included in the list if speed of call setup or takedown is a significant factor.

Alternate and Back Routing--Alternate routing is that network process wherein a node interrogates alternative nodes during line switched call setup. Back routing occurs when all alternate routes have been attempted and control of the routing must be returned to a previous node. These processes are best described with an example given in Figure 51.

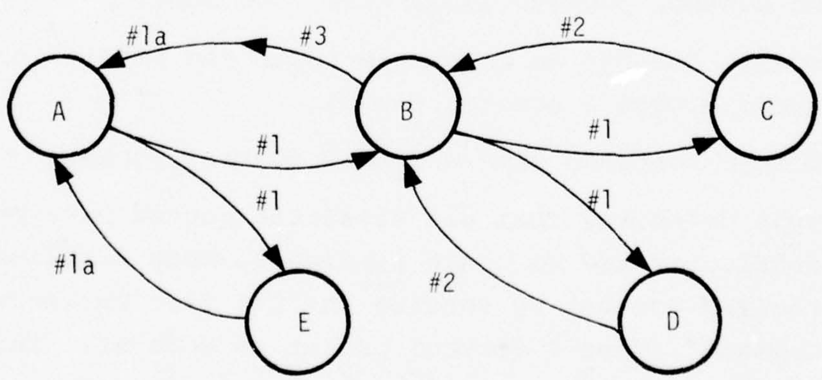


Figure 51. Alternate and Back Routing

Assume that a call is routed from Node A to Node B and that routes to Nodes C and D are unavailable. Alternate routing will be attempted by Node B. Routing control will be back routed to Node A where alternate routing will occur. These processes are implemented as follows:

- Node A transmits an "L/S Slot Reservation Command/ Request" (Type 1 control packet) to Node B.
- Node B responds with an "L/S Slot Reservation Command" (Type 1a control packet), thereby completing the bidirectional path between A and B. Control of the routing process now resides at Node B.
- Node B attempts to route the call to Node C by sending a Type 1 control packet.
- Node C is unable (or unwilling) to accept the call. It responds with an "L/S Slot Reservation Request Denied" (Type 2 control packet). Note that the route to Node C might also have been disqualified by Node B. In that case, the total effect would be the same but no control packets would have been sent.
- Node B selects an alternate route (to Node D) and sends a Type 1 control packet.
- Node D responds with a Type 2 control packet.
- Upon detecting that all alternate routes have been considered and none are available, Node B relinquishes routing control by sending an "L/S Slot Unreserve Command" (Type 3 control packet to Node A). This causes the removal of the path between Nodes A and B.
- Node A selects an alternate route (to Node E) and sends a Type 1 control packet.

- Node E responds with a Type 1a control packet, thereby completing the bi-directional path between A and E. Control of the routing process continues from Node E.

Packet Routing

At a node, packet routing is a three-part process (Figure 52). The process consists of:

- Selection for routing
- Route selection
- Selection for transmission

Selection for Routing--The description of all correctly received packets are searched to locate the packet in the receive queue which has the smallest delay margin and the highest precedence. Note that a low-precedence packet which is near its transmission deadline may be selected ahead of a higher precedence packet which has a large delay margin. This prevents low precedence packets from becoming stranded in a queue. Packet age and precedence are important selection parameters.

Route Selection--The chosen packet is routed by considering both the predicted delay to the destination and the local deterministic factors. Route selection criteria include:

- Queue lengths (delay) and capacities
- Precedence of existing L/S and P/S
- Line or trunk occupancy vs. capacity
- Line or trunk length (delay)
- Deterministic factors
- Predicted transmission time

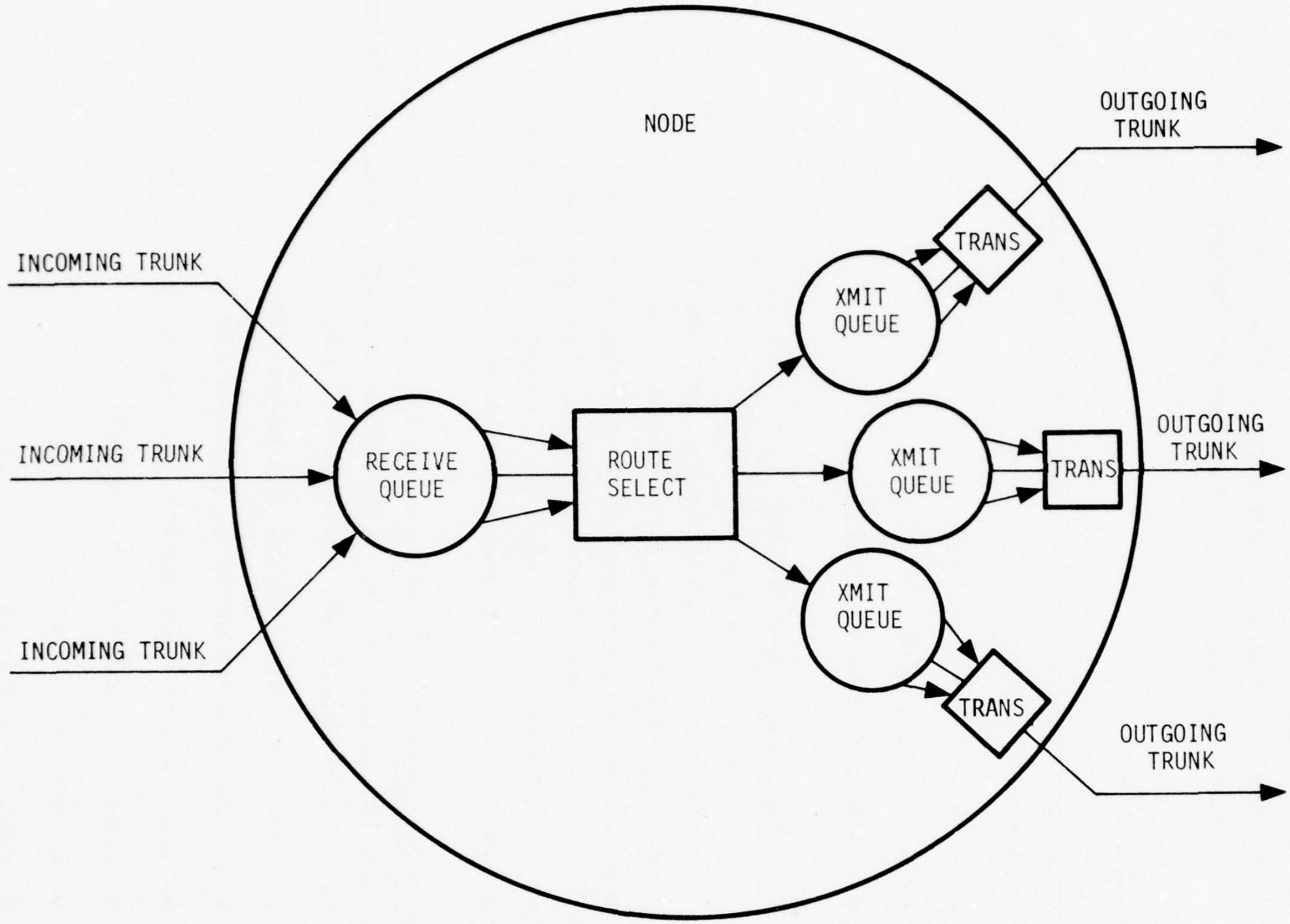


Figure 52. Packet Routing Process

After the route is selected, the packet is placed in a transmit queue dedicated to the outgoing trunk.

Selection for Transmission--The transmit queues are searched concurrently and a packet from each is selected for transmission on the corresponding outgoing trunk. Packet age and precedence are, again, important selection parameters. Selecting packets a second time permits precedence effects at each point of packet intersection.

SECTION 4

APPLICATION OF ARCHITECTURES

This section, based on the results covered in Section 3, presents a nodal data path structure, a selected associative architecture for L/S call functions, and a hardware structure for P/S call functions. For comparison purposes, this section also presents an alternate, and less attractive, associative architecture.

NODE DATA PATH STRUCTURE

Based on the definition of the integrated network functional design, it has been possible to describe a nodal data path structure consistent with that design (Figure 53).

The principal components include a fixed-delay switch, packet storage, and node control. (Note that this figure does not necessarily represent a particular implementation but rather only a possible data path structure.)

The fixed-delay switch provides for internal routing of line-switched (incompressible) data. The packet storage areas are used for reception, switching, and transmission of packet switched data. Separation by storage function permits the use of several memory technologies, thereby distributing and balancing the requirements for storage access.

The node control supervises the operations of the node. It formulates and maintains the transmit and receive tables which are the

PRECEDING PAGE BLANK-NOT FILMED

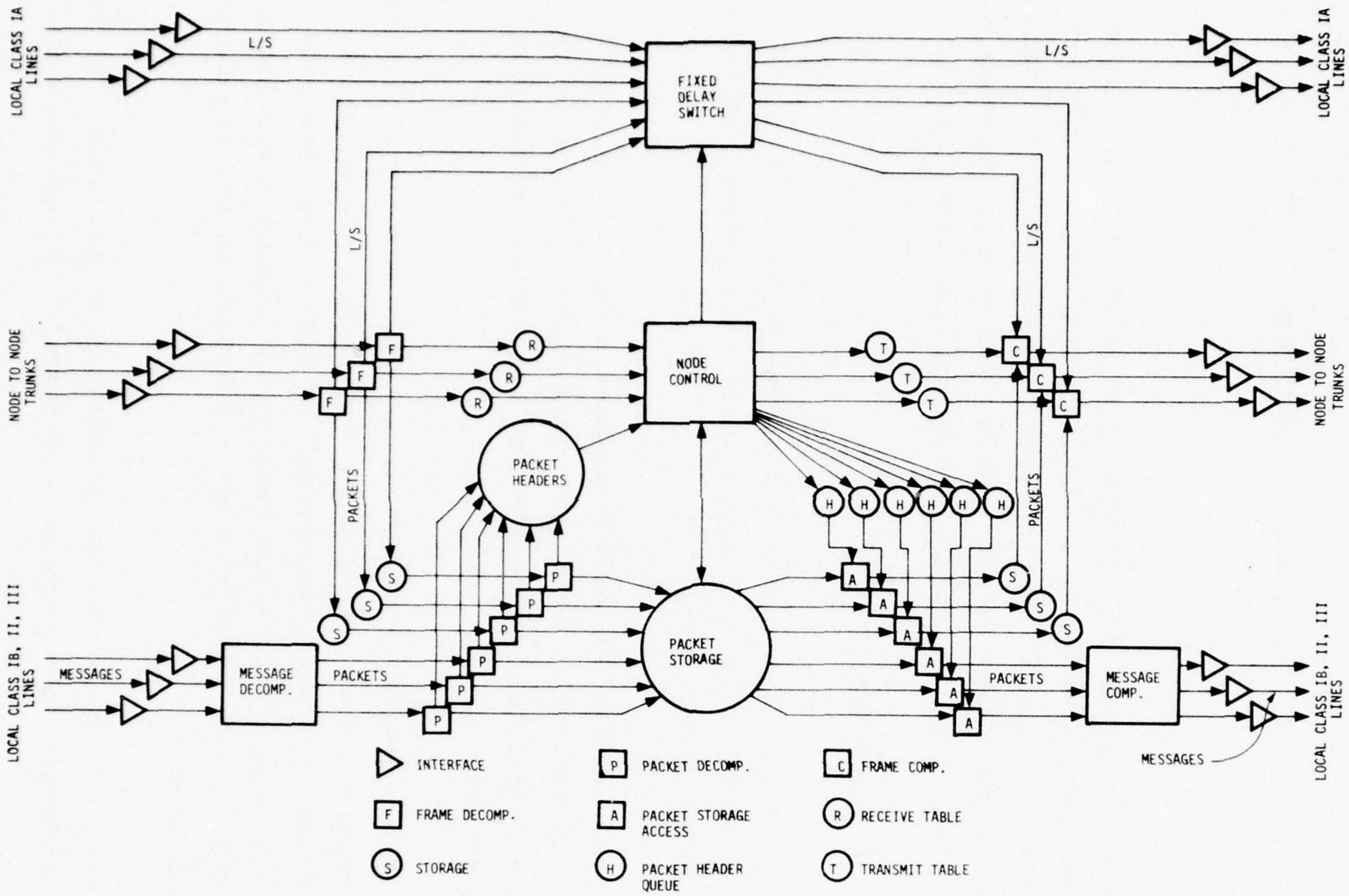


Figure 53. Nodal Data Path Structure

basis for composition and decomposition of frames on the node to node trunks. It controls the fixed-delay switch and the allocation of packet storage. The node control also monitors and controls the line and trunk interface circuits.

Frames received on incoming node-to-node trunks are decomposed into packet and line switched data. Decomposition control is provided by receive tables maintained by node control. Line-switched data is applied to the fixed-delay switch for time and space switching. Packet-switched data is stored until a complete packet has been received. The packet is then further decomposed by extracting its header for analysis. The remainder of the packet is stored until it is to be re-transmitted. Packets selected for re-transmission are moved to a storage area associated with the selected outgoing trunk. Transmit tables maintained by node control are used to compose line-switched data from the fixed-delay switch and packet-switched data from storage into frames which are transmitted on outgoing node to node trunks.

Local access is provided separately for line switched and packet switched data. Local line-switched data accesses are connected to the fixed-delay switch. Line concentration and expansion (not shown) can be used to optimize switch use. Local packet-switched data accesses provide buffer storage which is connected to the packet-switching facilities of the node.

ASSOCIATIVE CALL-BUFFER SWITCH FOR L/S CALLS (CLASS IA)

The Associative Call-Buffer switch was developed and designed to implement the L/S call (Class IA) switching function at each node in the network. As implied by its name, the Associative Call-Buffer switch employs associative techniques to perform the L/S

call switching operations. It also has special switching implementation that uses one individual buffer for each L/S call passing through the node. Figure 54 shows the block diagram for an Associative Call-Buffer switch at a node (and two closely related functional units, the frame decomposer and the frame composer, for each trunk).

Figure 54 shows the node hardware which implements frame decomposition, L/S data switching, and frame composition. On the receiver side, the frame decomposer and data distributor (FDDD) for each trunk performs these operations:

- distributes the received Frame Header and Sync Field (FHSF) into the FHSF input buffer for further processing
- distributes the packet data (including all four packet types, such as data, control, ack, and voice) into the P/S input buffer for further processing
- associatively distributes L/S data (Class IA) into individual FIFO buffers where each FIFO buffer has been assigned to one L/S call.

On the transmit side in Figure 54, the reverse operation occurs.

The L/S data fixed-delay switching is accomplished by: 1) temporarily storing the L/S data for each L/S call (from the incoming trunk) into the assigned FIFO buffer (which is sized for 10ms of data); and 2) unloading the FIFO buffer onto the appropriate output trunk at the appropriate time. There is an associative memory receive (transmit) table in each frame decomposer and data distributor (data collector and frame composer) unit. The associative memory transmit/receive tables are not only used for frame composition/decomposition, but also for containing the information necessary

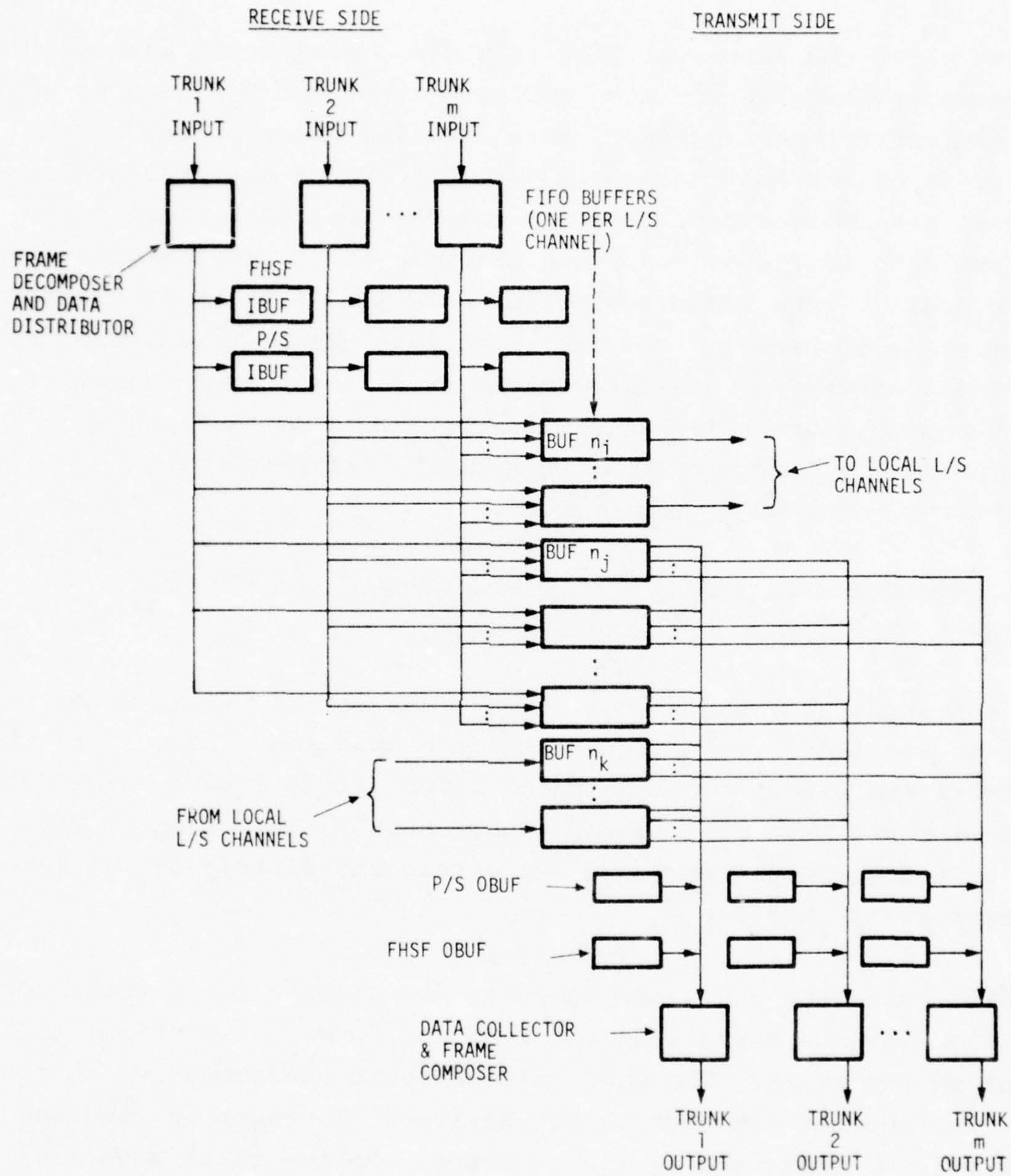


Figure 54. Node Data Flow -- Decomposition, Switching, Composition

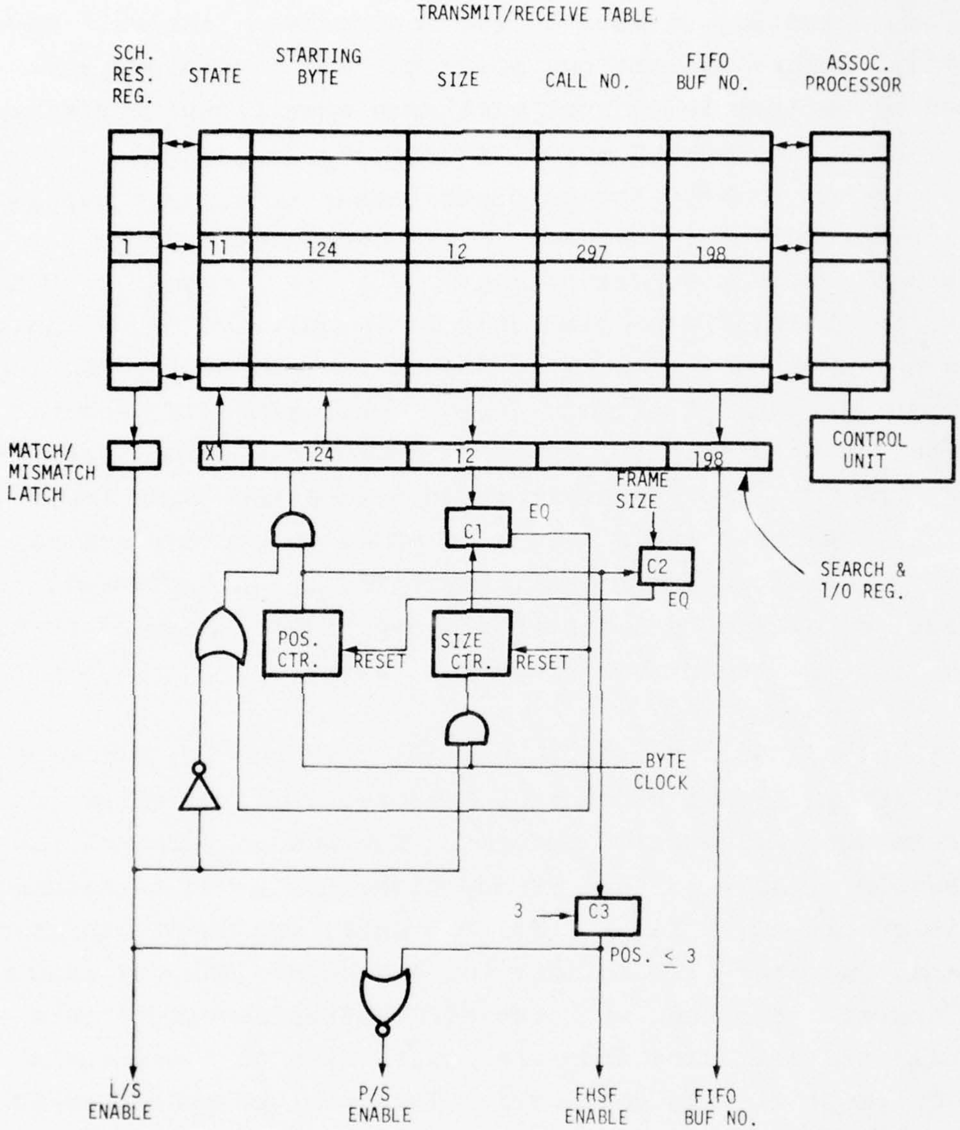
to accomplish the line-switching function. Additional associative processing techniques are also employed to switch L/S data to and from the correct call buffer. More detailed descriptions of the operations of the Associative Call-Buffer switch are presented in the next two subsections, "Frame Decomposer and Data Distributor Unit and Data Collector and Frame Composer Unit" and "Associative Gating Logic". The third subsection describes how the AM transmit tables are also used for L/S call slot reservation/de-allocation. The fourth subsection presents the hardware nodal requirements for the associative call-buffer switch. The last subsection, "Associative Call-Buffer Switch Features", describes the significant features of this switch.

Frame Decomposer and Data Distributor (FDDD) Unit and Data Collector and Frame Composer (DCFC) Unit

When the input data bit stream for a trunk enters into a node, it is first verified by the trunk interface unit (not shown in Figure 54) for frame synchronization check before it is input to the Frame Decomposer and Data Distributor (FDDD) unit of that trunk. The FDDD unit decomposes the input bit stream and distributes it into the appropriate buffers.

The Data Collector and Frame Composer (DCFC) unit for a trunk collects data from various buffers in the node and forms a bit stream to be output on the trunk. The DCFC unit performs operations which are opposite those of the FDDD unit. However, the controls required for these two units are similar. Hence, the two units have the same block diagram, as shown in Figure 55.

Each FDDD or DCFC unit contains an associative memory table, a simple associative processor (AP), an AP control unit, and control



FRAME COMPOSITION/DECOMPOSITION AND ASSOCIATIVE CALL-BUFFER SWITCH CONTROL

Figure 55. Frame Composition/Decomposition and Associative Call Buffer Switch Control

logic for generating various buffer load/unload enables. The associative memory and the control logic are similar to that proposed by Zafiropulo [6] but with some significant modifications: 1) the FIFO buffer number which is uniquely associated with one L/S call is not decoded into a great number of control signals (one line per buffer); instead, it is propagated down to the distributed associative gating logic by a small number of buffer address lines; and 2) the slot size is in units of bytes instead of bits and thus more time is available for processing the generation of enable control signals (i.e., 8-bit times vs. one-bit time in Zafiropulo's approach). A simple associative processor (See subsection "L/S Call Slot Reservation/De-Allocation Using Associative Memory (AM) Transmit Table") is also added to perform L/S call slot reservation/deallocation operations, directly on a transmit table (vs. list processing by Zafiropulo's approach). Most of these differences will be further discussed later.

At the receive side, the enable signals (L/S enable, P/S enable and FHSF enable) in Figure 55 control the gating of the trunk input data bit stream into appropriate buffers. One and only one of these three enable signals is "on" at any time. As shown in Figure 54, there is one input buffer for all P/S data, one input buffer for FHSF data, and many FIFO buffers for L/S data. The L/S enable signal operates together with the FIFO buffer number to gate a particular L/S slot data from the input frame to a particular FIFO buffer corresponding to that call. This will be explained further in the subsection "Associative Gating Logic".

As discussed in the L/S call setup/takedown procedures, each row in a receive (or transmit) table corresponds to a slot in the frame. A row with state 11 or 01 represents a L/S data slot, and only one of such rows may match in any one equality search operation, which

generates the L/S enable signal (that, in turn, affects the P/S enable signal as shown in Figure 55). Only the L/S data slot rows can affect the enable signals.

Refer to the associative memory table in Figure 55. In addition to the 2-bit state, each row has four entries: slot starting byte number (11 bits); slot size in bytes (11 bits); L/S call number (14 bits); and L/S call FIFO buffer number (12 bits). The L/S data slot rows in the table are created or removed during the setup or takedown of a L/S call on that trunk. Refer to Figure 5 for frame format. In the last portion of a frame, only packet data can be transmitted. During the transmission time for this portion of frame, the enable signals remain unchanged. This time can be used to update the associative memory table to reflect a new (old) L/S call being setup (takedown). In the transmit side, this time can also be used to process the L/S call reservation/ de-allocation procedures (using the associative transmit table and the simple associative processor). This is different from Zafiropulo's approach, which uses list structures within a conventional memory to process a L/S call slot reservation/ de-allocation on the frame. The L/S call slot reservation algorithm using an associative memory will be described in subsection "L/S Call Slot Reservation Using an Association Memory (AM) Transmit Table".

In a FDDD unit (Figure 55), there are two counters and three comparators in the control logic. The position counter has the current byte position number in a frame. For a 10-msec frame of a T1 line trunk, the position counter counts from 0 through 1929 (there are 1930 bytes or 15,440 bits in one 10-msec frame). It is reset to zero when it reaches the frame size (1930) by the output of Comparator C2. The size counter keeps the number of bytes, within a L/S slot, which have already been processed (i.e., already gated

into a FIFO buffer from a trunk input stream). It is reset to zero whenever the complete L/S slot data have been processed. Both the position counter and size counter are stepped by the byte clock (not the bit clock). The functions of the three comparators, C1, C2, and C3, are clearly illustrated in Figure 55.

The control logic and associative memory table generate the various enable signals as described in the next few paragraphs.

Both position counter and size counter are initially set to zero. The 1-bit match/mismatch latch register is also zero (L/S disabled) initially. The content of the position counter is gated into the starting byte field of the search and I/O register. The second bit of the state field in that register is set to one. (The first bit of the state field is "don't care", since both 11 and 01 states represent L/S slot.) The associative memory table is searched on the second bit of the state field and the complete starting byte field to see if any row has content in those bit positions matching those in the search and I/O register.

Because the first four bytes in every frame are FHSF, no L/S slot can start at Byte 0. Hence, the first equality search with zero on the starting byte field will result in no response, and the match/mismatch latch register remains zero after the search. Therefore, the L/S enable signal is still zero (i.e., L/S disabled). Similarly, the next three searches on the byte positions one, two, and three will result in no match and the L/S enable signal remains zero.

The value of the position counter is always compared to a constant "3" by the comparator C3. FHSF Enable will be 1 if and only if the position counter has a value of 0 through 3. Therefore, during the

time for the first four bytes of a frame, FHSF enable is one (on). The other two enable signals are zero.

When the value in the position counter advances to "4", then the FHSF enable is no longer "1". If the slot starting at the fourth byte is not a L/S slot, then the equality search on the AM will still result in no match and the L/S enable remains zero. With both FHSF enable and L/S enable being zero, the P/S enable becomes "1" as indicated by the logic diagram.

The equality search operation is executed once every byte time until a match results. At that time, the AM row containing the starting byte number of a L/S slot will respond to the search with a match. Its corresponding bit in the search results register is set to "1" and the match/mismatch latch register becomes "1", which is the value of the L/S enable.

After a successful search with a match, the size field and the FIFO buffer number field of the matching AM row are read out to the search and I/O register. The FIFO buffer number together with L/S enable is broadcast down to the associative gating logic to properly gate the L/S data into a pre-assigned FIFO buffer.

Once the match/mismatch latch register has value "1", the byte clock is able to advance the size counter. The value of the size counter is constantly compared by comparator C1 with the value stored in the size field of the search and I/O register. When data of a complete L/S slot have been gated into the FIFO buffer (the two inputs to comparator C1 are then equal), the equality search operation on the starting byte and state fields will be re-activated by the output of comparator C3. At that moment, the position counter contains the next byte position number in the frame.

The above operations are repeated continuously. When the value of the position counter reaches the frame size (1930), it is reset to

zero by the output of comparator C2; and the processing for the data of a new frame is started.

The operations in the Data Collector and Frame Composer (DCFC) unit at the transmit side are similar to those described above.

Associative Gating Logic

As mentioned in the previous section, at any given byte time, the FDDD unit at the receive side of each trunk has three different enable control signals and a FIFO buffer number available to control the gating of input bit stream into various buffers. The interfaces between these control lines, the FIFO buffer number lines, the trunk input bit stream lines and the buffers are shown in Figure 56. (The counterpart gating logic for the transmit side of each trunk is shown in Figure 57). The gating logic enclosed by the dotted lines in Figure 56 or 57 is called the associative gating logic.

In Figure 56, the FHSF enable signal of each trunk enables the gating of the first four bytes (which is FHSF data) of each input frame from the trunk into its FHSF buffer for further processing. Similarly, the P/S data enable signal of each trunk enables the gating of bytes in the P/S slots of each input frame from the trunk into its P/S buffer for further processing.

The data in the L/S slots of a trunk pass through an AND gate controlled by the L/S enable signal of that trunk. These L/S data are then broadcast down to the associative gating logic to be gated into appropriate FIFO buffers. The use of the FIFO buffer number sent out by each FDDD unit to control the gating of L/S data into the appropriate FIFO buffers is explained below. As shown in Figure 56,

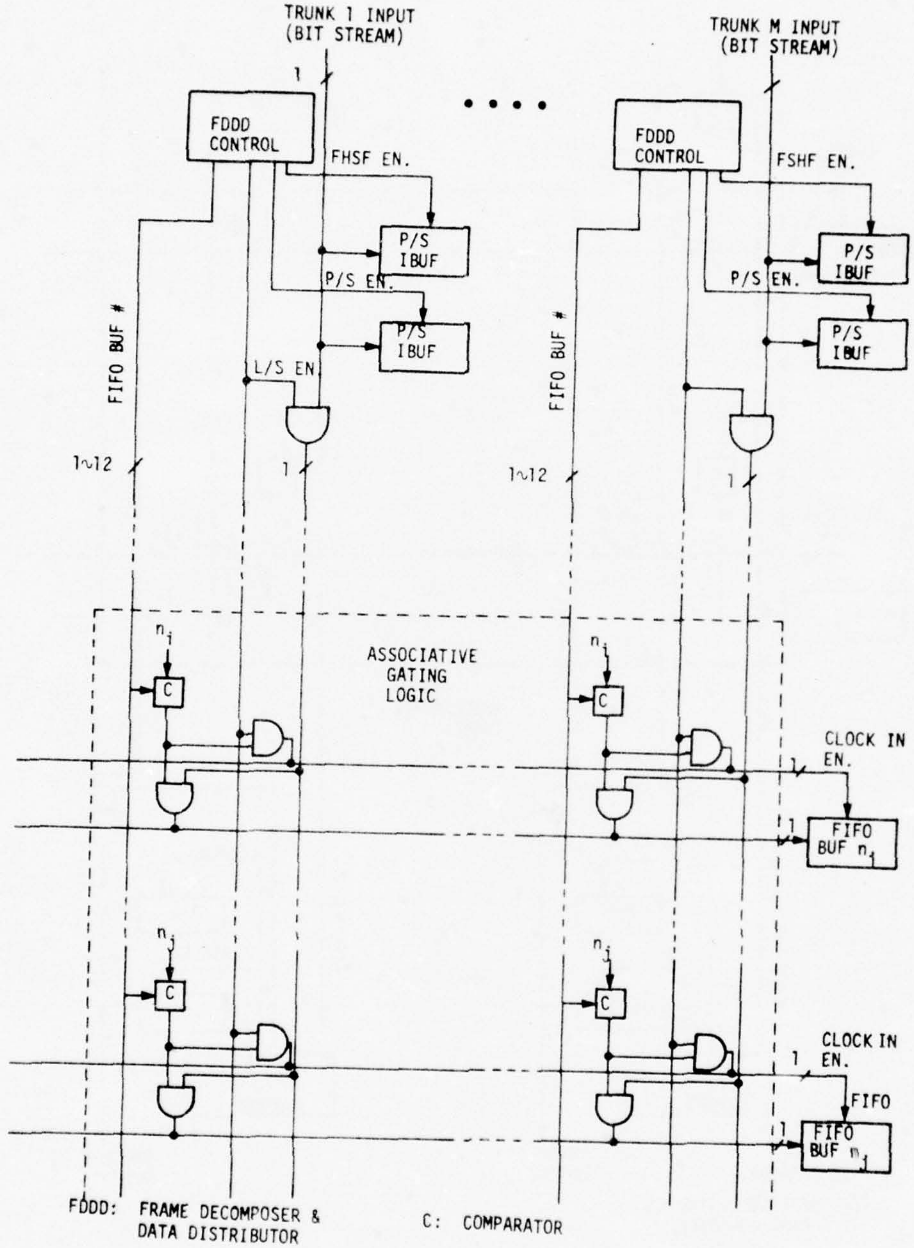


Figure 56. Associative Call-Buffer Switch Receive Side Logic

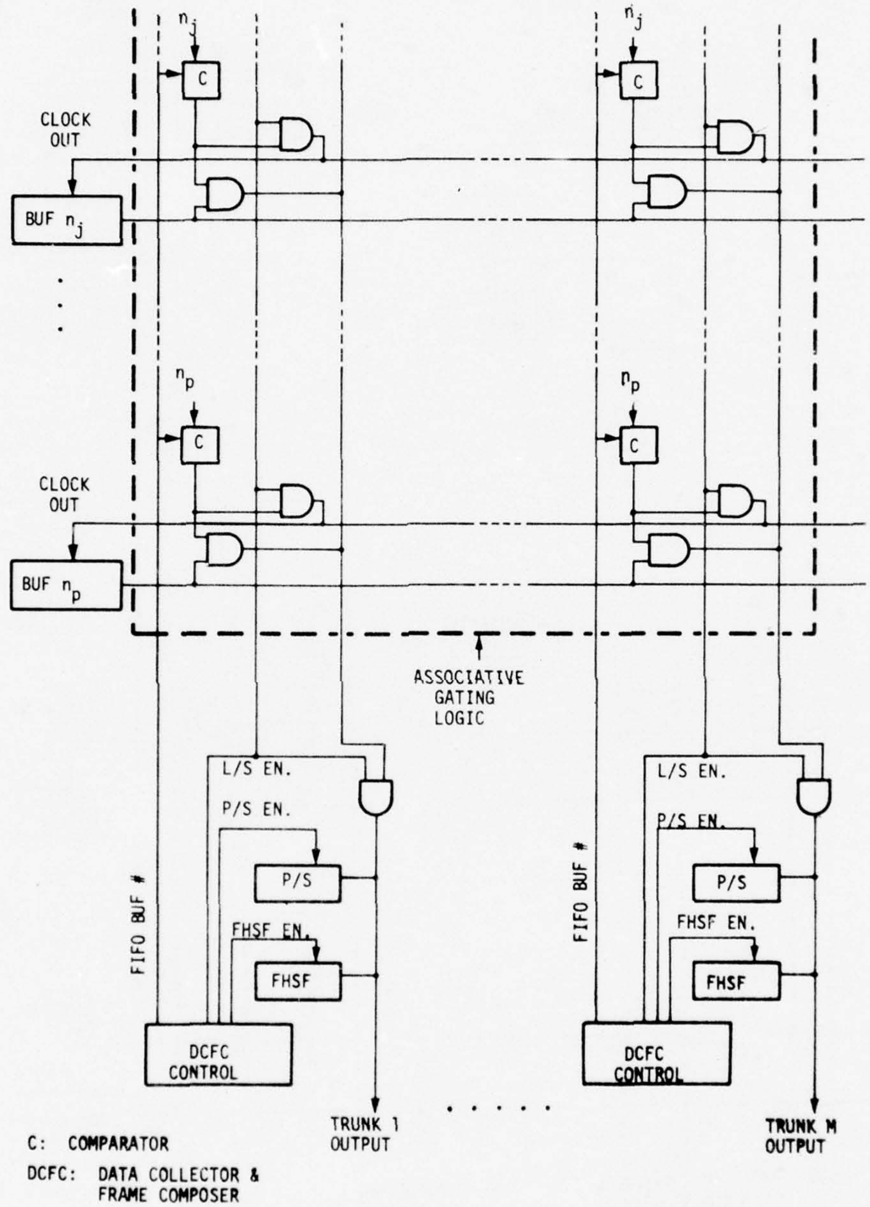


Figure 57. Associative Call-Buffer Switch Transmit Side Logic

each FIFO buffer has a number which is fixed and unique to the node. This FIFO buffer number (e.g., n_i in the figure) is constantly compared with the FIFO buffer numbers sent out by all FDDD control units. Whenever a match (equal) occurs, the L/S data of the trunk whose FDDD unit has sent the matching FIFO buffer number are gated into that FIFO buffer.

As mentioned earlier, each FIFO buffer corresponds to a unique L/S call passing through the node. Each FIFO buffer can be assigned to a L/S call of any trunk of the node during the setup of an L/S call. Once it is assigned, it is exclusively used by that call until the termination of that call. When the call terminates, the buffer becomes free again and it can be reassigned to a new L/S call from any trunk. In Figure 56, the comparators in a row for a FIFO buffer are comparing the various buffer numbers (sent down by the FDDD Units) to a fixed FIFO buffer number (e.g., n_i), and at most one of them can have a match at any time. The comparator that matches will continue to be the same one until the FIFO buffer on this row is reassigned to a new L/S call.

Also, among the comparators in a column for a trunk, at most one of them can have a match at any time. (Each FIFO buffer has a unique buffer number and at most one of the FIFO buffer numbers in the column can match the FIFO buffer number sent down by the FDDD unit of a trunk).

As shown in Figure 56, one of the two inputs to each comparator is the fixed number for the FIFO buffer in that row (e.g. n_i). This number can be preset to the predetermined FIFO buffer number of the FIFO buffer in that row during the construction of the associative gating logic.

The output of a comparator in the associative gating logic is "1" (i.e. on) when its two inputs are equal. Refer to Figure 56, the output of each comparator is fanned out to two AND gates, one ANDed with the L/S enable signal from the FDDD unit of the trunk, and the other ANDed with L/S data that is broadcast down from the same FDDD unit. The output of the second AND gate is the L/S data bit (one or zero) from the input data stream to be stored into the FIFO buffer. The output of the first AND gate is used as an enable to "clock" the data bit into the FIFO buffer. The enable signals across a complete row are wire-ORed together to form the resultant "Clock In" enable signal for the FIFO buffer in that row. The data bits across one complete row are also wire-ORed together to form the resultant input bit to be clocked into the FIFO buffer. As mentioned earlier, in the whole row of comparators, at most one of them has "one" as its output.

A preliminary study indicates that the number of FIFO buffers at a node will not exceed 1290 (see subsection "Associative Memory (Tables) and FIFO Buffer Requirements"). Hence, the FIFO buffer number will not exceed 12 bits ($2^{12} = 4096$). Because the "compare" operation is executed once each byte time (which is 5.2 μsec for T1 line rate), a sufficient amount of time is available for the comparator to get its input ready. Therefore, the 12-bit FIFO buffer number may be input in serial fashion to the comparator. This will reduce the number of interconnection lines between the FDDD units and the associative gating logic.

The associative gating logic in the transmit side (shown in Figure 57) is similar to that in the receive side discussed above, but it performs the reverse operations. That is, it associatively gates L/S data out from the FIFO buffers to the output trunk data stream lines.

L/S Call Slot Reservation/De-Allocation Using Associative Memory (AM) Transmit Table

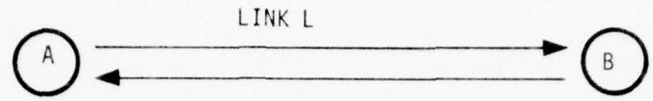
The L/S call slot reservation is the first of the two steps for setting up a L/S call. (The other step is L/S call slot allocation, as explained in subsection "Normal Line-Switched Call Setup". The L/S call slot de-allocation is the second of the two steps for taking down a L/S call. (The first step is L/S call slot de-reservation, discussed in subsection "Normal Line-Switched Call Takedown".) The L/S call slot reservation at a transmit side is to find best-fit available (P/S) slot(s) in the frame on the selected transmission trunk for the L/S call data, and then update the transmit table accordingly. The L/S call slot de-allocation performs the reverse operations (i.e. it returns L/S slots to P/S slots). Generally, new rows are created for new L/S slots in the AM transmit table during the L/S slot reservation process; and the rows representing the L/S call being taken down are removed from the AM transmit table during the L/S call slot de-allocation process. The other two steps in L/S call setup/takedown--L/S call slot allocation and L/S call de-reservation--only modify the state field of related rows in the AM transmit and receive tables.

The AM transmit/receive tables discussed earlier only require the information for L/S data slots (State 11 or 01). Such information is sufficient for frame de-composition and data distribution in the receive side and for data collection and frame composition in the transmit side. But, as explained below, the AM transmit table with some expansion can also be used to process the L/S call reservation/de-allocation operations.

As shown in the network frame format in Figure 5, all Class I calls (which includes all L/S calls and packetized voice calls) are limited

to the portion of the frame before the dotted line (L/S Limit). The portion of the frame beyond the dotted line is reserved for packet data for high priority control packets. During the time of transmitting the portion of frame data beyond the dotted line, the transmit table is not actually used. (The enable signals generated by the DCFC unit remain unchanged during that period). During this period, the transmit table, with some expansion, can be used to process the L/S call reservation/de-allocation operations. The length of this period is at least 186 μsec which is equivalent to 288-bit times on T1 line), since it must provide trunk space for at least one control packet which can be 288-bits long. This time (186 μsec or longer) is sufficient for executing the L/S call reservation/de-allocation operations. In the following two subsections, the details of the L/S call reservation and de-allocation algorithms and their operations will be described.

<u>L/S Call Slot Reservation Using An Associative Memory (AM) Transmit Table</u> -- To set up a L/S call through a node, the node will first select an output trunk (route) and then reserve slots for the call on the frame of that trunk. The AM transmit table of each trunk must now contain information for both L/S slots and non-L/S slots (which are also simply called P/S slots, i.e., they are to be filled-in with P/S data or idle characters), so that best-fit non-L/S slots can be identified and reserved for the new L/S call. A typical AM transmit table is shown at the left half of Figure 58. Note that the rows with state 00 or 10 correspond to P/S slots and that they do not respond to the Equality Searches during the data collection and frame composition. The rows with state 10 correspond to slots that are transmitioning to L/S slots. The rows with state 00 correspond to free P/S slots that can be reserved for new L/S calls. Note that the portion of frame beyond the dotted line in the frame input is reserved for P/S data only and, hence,



	STATE (2 BITS)	STARTING BYTE (11 BITS)	SIZE (11 BITS)	CALL NO. (14 BITS)	FIFO BUF # (12 BITS)
P/S	00	35	1855		
L/S	11	4	6	87	251
P/S	00	20	2		
L/S	11	22	3	9	190
P/S	10	10	10	38	7
L/S	01	25	10	129	76

TRANSMIT TABLE AT NODE A
FOR LINK L

	STATE	STARTING BYTE	SIZE	CALL NO.	FIFO BUF NO.
L/S	11	4	6	87	95
L/S	11	22	3	9	154
P/S	10	10	10	38	637
L/S	01	25	10	129	2

RECEIVE TABLE AT NODE B
FOR LINK L

Figure 58. Transmit/Receive Table
(Associative Memory)

cannot be included in the transmit table as part of a free P/S slot. The right half of Figure 58 is a typical AM receive table. The receive table contains only rows corresponding to L/S slots (which has state 11 or 01) or P/S slots that are in the transition state (01) to become L/S slots. It does not have rows with state 00. Of course, there must always be some spare rows in the AM transmit/receive tables so that additional rows can be created to describe the new L/S call data slots when a new L/S call is to be set up. The simple associative processor shown in Figure 55 (beside the transmit/receive table) contains Effective Row (ER) tags (1 tag bit per AM word) which indicate which rows are spares and which rows describe the effective slots in the frame.

To set up a L/S call at a node, a FIFO buffer (in the Associative Call-Buffer Switch) must be reserved for that call. There are three types of FIFO buffers (Refer to Figure 54): local L/S channel to network trunk, network trunk to network trunk, and network trunk to local L/S channel. Each type may have buffers of various sizes to accommodate various L/S call data rates. When the FIFO buffers are divided into several categories by their types and sizes, a FIFO buffer availability table is required for each category to keep the record of free FIFO buffers for each category. If a new L/S call of that category is to be set up, then an appropriate free FIFO buffer can be found and assigned to the L/S call.

The procedure to reserve free P/S slot(s) for a L/S call works as described in the following paragraphs:

If the width of the call (i.e. L/S call data size for one frame time) is less than or equal to the largest free P/S slot size, then find a best-fit free P/S slot and reserve the whole slot (if the fit is exact) or a portion of it (if the fit is not exact) for the L/S call. For the former case, the state, the call number field, and

the FIFO buffer number field of the row for the best-fit slot will be updated to be 10, L/S call number and assigned FIFO buffer number, respectively. For the latter case (i.e. not exact fit), the found P/S slot is divided into two slots: one is the new L/S slot, the other is the remaining free P/S slot (smaller than before). Therefore, a new row representing the slot for the L/S call must be created from spare rows in the AM transmit table. The row representing the old P/S slot must be updated in its slot starting position and size to represent a new smaller free P/S slot.

On the other hand, if the width of the L/S call is greater than the greatest free P/S slot, then the greatest free P/S slot is reserved for the L/S call (by updating the various fields of row representing it) and the remainder width of the L/S call is computed. The reservation process described above then repeats for the remainder width. This procedure continues until no remainder width exists. (It is assumed that, before the output trunk is selected for this L/S call, the total available free P/S slot space has been checked to make sure that there is sufficient free trunk space for this new L/S call).

The L/S call slot reservation procedure described above is outlined in Figure 59.

All operations for L/S call slot reservation can be carried out by the simple Associative Processor and the AP control unit provided for each AM transmit table as shown in Figure 55. They are executed during the time of transmitting the portion of frame data beyond the dotted line (L/S Limit) in the frame format (when the transmit table is not used for data collection or frame composition).

L/S CALL RESERVATION USING TRANSMIT TABLE (ASSOCIATIVE MEMORY)

(1) ANY P/S SLOT SIZE $\geq$ (REM.) L/S CALL WIDTH? IF NO, GO TO (8).

(2) FIND BEST-FIT P/S SLOT FOR (REM.) L/S CALL WIDTH.

(3) BEST-FIT P/S SIZE = (REM.) L/S CALL WIDTH? IF YES, GO TO (6)

(4) MODIFY THE P/S SLOT ROW: SLOT STARTING POSITION, SIZE.

(5) ACTIVATE A NEW AM WORD TO REPRESENT A SLOT RESERVED FOR THE L/S CALL: STATE (10), SLOT STARTING POSITION, SIZE, L/S CALL #, FIFO BUF #
GO TO (7).

(6) MODIFY THE P/S SLOT ROW: STATE (10), L/S CALL #. FIFO BUF #

(7) END.

(8) FIND MAX. SIZE P/S SLOT AND MODIFY IT TO REPRESENT A SLOT FOR A PORTION OF THE L/S CALL:
STATE (10), L/S CALL #, FIFO BUF #

(9) UPDATE (REM) L/S CALL WIDTH AND GO TO (1).

Figure 59. L/S Call Reservation Using Transmit Table (Associative Memory)

<u>L/S Call Slot De-Allocation Using an AM Transmit Table</u> -- The L/S call slot de-allocation process is the reverse process of the L/S call slot reservation. It returns L/S slots to P/S slots. The new P/S slots are then combined with adjacent existing free P/S slots (if any) to form new larger-size free P/S slots. If the combination of slots occurs, the total number of slots (including both L/S and P/S) will be reduced. Hence, the total number of effective rows describing the frame may descrease.

There is no "find the best-fit slots" operation in the L/S slot de-allocation process. Instead, the major operations in the L/S slot de-allocation process are identifying the L/S slot row(s) for the terminating L/S call in the transmit table and then either just changing their state to the P/S state (if the existing adjacent

slots are not free P/S slots) or combining them with existing adjacent free P/S slots. The combing process involves updating the starting byte field and size field of existing free P/S slot rows and removing those rows from the transmit table which no longer describe slots of the frame. (A row in the transmit table is considered removed when its Effective Row bit in AP is set to 0, i.e., it becomes a spare row.) Similar to the L/S call slot reservation operations, all of these operations can be carried out by the simple associative processor and the AP control unit which are provided for each AM transmit table as shown in Figure 55. They are also executed during the time of transmitting the portion of frame data beyond the dotted line in the frame format.

Associative Memory (Tables) and FIFO Buffer Requirements

The major components for handling the Class IA (non-clear-voice) calls using the associative call-buffer switch at a node are the AM transmit tables (one per trunk), the associative gating logic at the transmit side, the AM receive tables (one per trunk), the associative gating logic at the receive side, and the FIFO buffers (one per L/S call through the node). The size and quantity requirements of these components are analyzed below based on some assumptions of the network size, ratio of Class IA (L/S) data vs. other types of data, data rate distribution among Class IA data, and average number of slots for a L/S call in a frame.

Assumptions:

1. 16 T1 trunks at a node
2. Maximum 85% Class I data on trunks
3. Among Class I data, 70% are Class IA (Non-clear-voice) and 30% are Class IB (Clear-voice)

4. Data rate distribution for Class IA data

   25% with rate 64 kbps ∿ 300 kbps
   50% with rate 9.6 kbps ∿ 56 kbps
   25% with rate 2.4 kbps ∿ 8 kbps

5. On the average, 1.5 slots in the frame for each L/S (Class IA) call

6. Frame size is 10 msec equivalent (i.e. 15440 bits with T1 rate)

Requirement Computations -- By (2) and (3) above, the Class IA data is at most about 60% (85 percent x 70 percent) of the total traffic. In one frame (for a trunk), the Class IA data will not exceed 9240 bits (15440 x 60 percent). Among the 9240 bits, 4620 bits (50 percent) belong to the group of medium data rates (9.6 kbps to 56 kbps), 2310 bits (25 percent) belong to the group of high data rates (64 kbps to 300 kbps), and another 2310 bits (25 percent) belong the the group of low data rates (2.4 kbps to 8 kbps). Note that an 8 kbps L/S call contains 80 bits in a 10-msec frame, etc.

To compute the number of Class IA calls for each data rate group on a T1 trunk, the further assumption is that the average data rate for each data rate group is 64 kbps for the high data rate group, 16 kbps for the medium data rate group, and 4.8 kbps for the low data rate group. The numbers of Class IA calls per T1 trunk for the three data rate groups are as follows:

No. of high data rate calls	=	2310/640	=	3.61
No. of medium data rate calls	=	4620/160	=	28.88
No. of low data rate calls	=	2310/48	=	48.13
Total				80.62

By assumption (5), the number of L/S data slots in a frame will be 121 (80.62 x 1.5) that is, 121 rows are required in the AM table to describe L/S slots. It is estimated that an AM table with 200 rows is sufficient to provide additional rows for P/S slots and for spare rows.

The word size of the AM memory is 50 bits (see Figure 58). Hence, each AM table is 200 words x 50 bits. Since there are 16 trunks at a node and each trunk has one transmit table and one receive table, 32 such AM tables are required at a node.

To compute the number and size of FIFO buffers required at a node, note that at maximum there are a total of 1290 Class IA calls (80.62 calls per trunk x 16 trunks) passing through the node. Because each Class IA call requires a FIFO buffer at the node, a total of 1290 FIFO buffers are required at a node. Each FIFO buffer needs only to store data of the call of one frame (since the data of a L/S call loaded into the FIFO buffer for that call will be output on some output trunk within one frame time.) Hence, the size requirements for the FIFO buffers depend on the data rate of the call that uses it. In the analysis below, the FIFO buffers are divided into three groups by size: 3000 bits (capable of handling a Class IA call of rate up to 300 kbps); 560 bits (for rate up to 56 kbps); and 80 bits (for rate up to 8 kbps). The number of FIFO buffers for these three groups required at a node are:

FIFO Buffer Size	Number of FIFO Buffers
3000 bits	58 (3.61 calls per trunk x 16 trunks)
560 bits	462 (28.88 " " x 16 trunks)
80 bits	770 (48.13 " " x 16 trunks)
Total	1290

The above computation of FIFO buffer requirements at a node assumes that the FIFO buffers have only three sizes (for the specified three data rate groups). Because any FIFO buffer in a group must accommodate the highest data rate call allowed in that group, all FIFO buffers in the group must have the maxiumu length. The waste of FIFO buffer storage is apparent. For instance, in the high data rate group, using a 3000 bit FIFO buffer for a 64K bps call wastes 2360 bits (3000-640) of storage. The waste of storage can be avoided to a certain degree by dividing the FIFO buffers into more groups, each with a smaller range of data rates. This will increase the processing complexity in assigning a FIFO buffer for a Class IA call during the call setup process. There is a tradeoff between the processing complexity and buffer storage. An extreme solution is to divide the FIFO buffers into groups where each group corresponds to one and only one data rate. The waste of buffer space is then kept to a minimum.

The associative gating logic at the receive side (or at the transmit side) is a two-dimensional array consisting of identical elements, where each element corresponds to an intersection between a trunk and a FIFO buffer. The circuitry in each element consists of a 12-bit comparator and two AND gates. For the assumed node size of 16 T1 trunks and 1290 FIFO buffers, each of the two arrays contains 16 x 1290 or 20640 elements. (A number of elements may be grouped together to form one module, as discussed later on the modularity feature of the Association Call-Buffer Switch).

Below is a summary of node requirements:

(1) Associative Memory Tables:
Number: 32
Size: 200 words x 50 bits

(2) FIFO Buffers (Assuming total FIFO buffers are divided into three groups and that buffers in each group have the same size):

FIFO BUFFER SIZE	NUMBER OF FIFO BUFFERS
3000 bits	58
560 bits	462
80 bits	770
	TOTAL 1290

(3) Two associative gating arrays: each array has 20640 elements; each element has a 12-bit comparator and two AND gates.

Associative Call-Buffer Switch Features

The block diagram, associative gating logic, operations and, requirements of the Associative Call-Buffer switch have been described in previous sections. The Associative Call-Buffer switch employs associative memories as receive tables and transmit tables in which switching information for L/S calls are stored. It also uses simple associative gating logic to accomplish switching of L/S calls through a node. In addition to providing the information for frame composition and L/S switching operation, the associative memory transmit tables are also used for setting up and taking down L/S calls (i.e. L/S call slot reservation and de-allocation processing).

Other features of the Associative Call-Buffer switch that are worth noting are described below:

Fixed Minimal Node Delay -- The L/S call data are loaded from incoming trunks into the FIFO buffers in real time (i.e. as they are entering the node). Meanwhile, the outgoing trunks are collecting the L/S call data from the FIFO buffers to form output frames and output them on the trunks in real time. After a L/S call is set up, the slots for the call on the incoming frame and the slots for the same call on the outgoing frame have a fixed time relationship. The time difference between incoming slots and outgoing slots is always less than or equal to one-frame time (e.g. 10 msec). This is so because a FIFO buffer is filled in at one end of the buffer with incoming data for that call and the data are taken out at the other end of the same buffer to the output trunk within one frame time. Because any switch that performs time switching must have at least this amount of delay through a node, the delay caused by the Associative Call-Buffer switch (which performs the complete L/S switching function) is therefore minimal. Hence, the associative call-buffer switch provides fixed, minimal node delay for the L/S calls. This feature results in minimal cross-network delays for the L/S calls.

Modularity and Expandability -- The modular feature of the associative call-buffer switch is illustrated in Figure 60, which shows the components at the receive side and the FIFO buffers. Each Frame Decomposer and Data Distributor (FDDD) unit for each trunk is a module. A group of associative gating logic elements may form a module. Whether the FIFO buffers can be grouped into modules depends on the sizes of the individual FIFO buffers. It is suspected that some of them can also be grouped into modules.

The requirement analysis in "Associative Memory (Tables) and FIFO Requirements" on page 142 shows that a T1 trunk passing through a node contains about 80 L/S calls (Class IA). Hence, if a new

trunk is to be added to a node, the modular expansion of the associative gating logic will expand much faster in the vertical direction than in the horizontal direction. This suggests that a module in the associative gating logic should correspond to one trunk and many FIFO buffers. Another factor from the associative gating logic element circuitry property also points toward this type of modularity; that is, each element has only two line connections in the horizontal direction (to a FIFO buffer) but has four or more line connections (to a FDDD unit) in the vertical directions. (The four or more vertical lines are one data line, one L/S enable line, one control line, and one or more of FIFO buffer number lines.) For a given number of pin connections, a module with elements corresponding to one trunk and many FIFO buffers will contain the greatest number of elements. As an example, a module can contain 1 x 20 or 20 elements and has a total of 44 pins (assuming four line connections in the vertical direction).

Figure 60 shows the case of a node expanding from four trunks to five trunks. A new FDDD unit is added. A number of FIFO buffers (e.g. 80 buffers in our example) of various sizes are added. The associative gating logic is expanded in two directions. It expands one column of elements in the horizontal direction and 80 rows of elements in the vertical direction. Using 20-element modules, a total of 5 x 80/20 + 4 x 80/20 or 36 modules are required to be added to the associative gating logic to expand a four trunk node to a five trunk node.

LSI Technology Feasibility -- The associative gating logic is suitable for LSI circuitry technology because: the logic is uniform and simple, the speed requirement is not critical, there

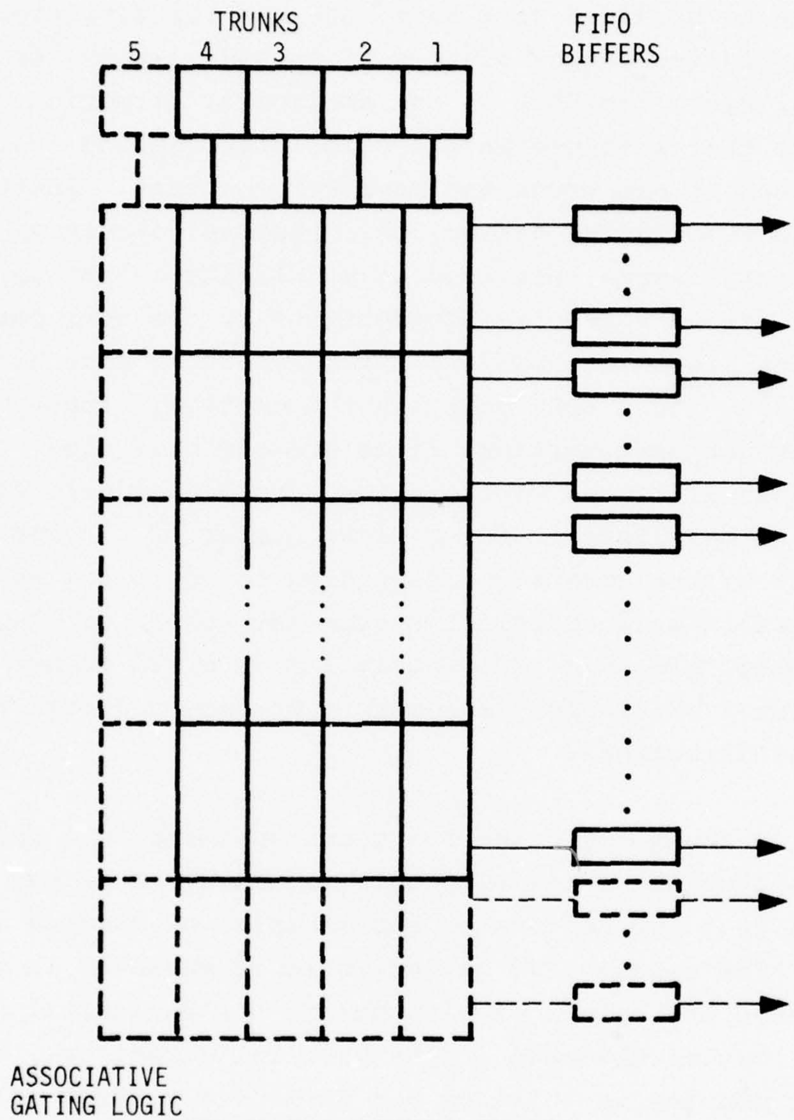


Figure 60. Modularity/Expandability of the Associative Call-Buffer Switch

is no substantial power consumption, and all modules in the associative gating logic are identical.

NODAL PACKET SWITCHING FACILITIES

A mechanism has been defined which performs frame decomposition, fixed-delay switching, and frame composition. In that definition packet-switched data was extracted for switching via a separate mechanism. The functional characteristics of that mechanism have been discussed in the previous "Associative Call-Buffer Switch for L/S Calls" on page 121. The current section presents and analyzes a hardware architecture which has been specifically tailored to the packet switching function.

Hardware Structure and Operation

Figure 61 represents the hardware facilities within a node for packet switching. Packet data streams are applied to the inputs and "Switched" packet data streams are taken from the outputs. Independent packet storage buffers (labeled S) are provided at each input and output. The storage buffers receive (or provide) data in accordance with the real-time requirements of frame decomposition and composition.

Incoming data packets are stripped of their header fields by the packet decomposition hardware, P. The packet headers are collected in a common storage area, PH, for analysis by the node control, NC, while the packet content is placed in a larger packet storage area, PS, to await re-transmission.

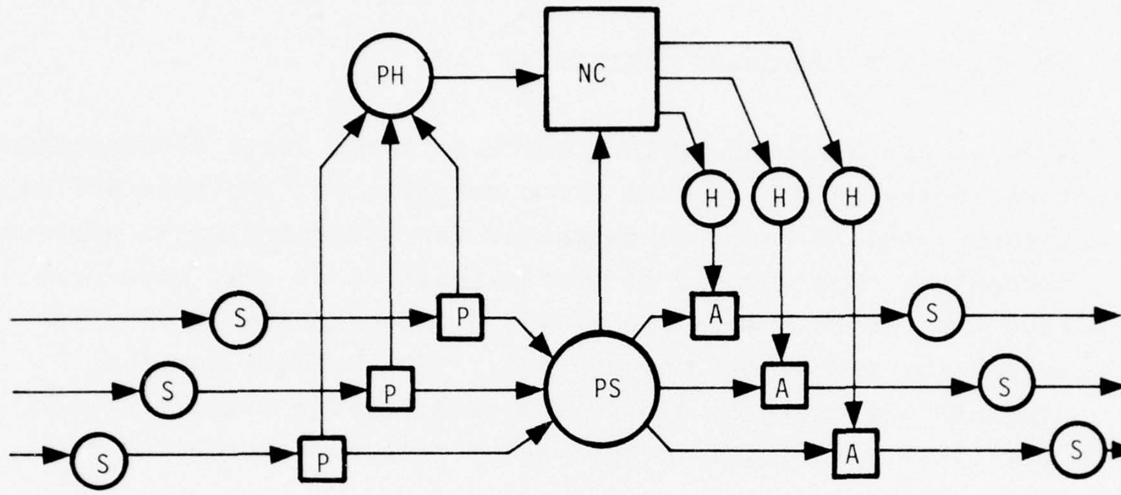


Figure 61. Packet Switching Facilities

As each packet header is processed by the node control it is placed in one of the outgoing header queues, H. Each header queue corresponds to a specific output trunk. The packet-access hardware, A, selects a packet header from its outgoing header queue, combines it with the data portion from the packet storage area, and places the

complete packet in the outgoing storage buffer. The complete packet is thus available for frame composition on the selected outgoing trunk.

It is not necessary to actually perform the packet decomposition/ composition in the manner described above. Complete packets may be retained in the packet-storage area while a translated form of the packet header information is processed by the node control. The packet-access hardware then would not have to perform a composition function. The advantages and disadvantages of this approach are not readily determined and are best evaluated with respect to an actual hardware implementation.

Memory Requirements

It is possible to perform an initial analysis of the memory requirements imposed by the packet-switching hardware structure which has been presented.

If T1 trunks and a 10ms frame are assumed, each input storage buffer will receive a new bit each 647 ns. If higher bandwidth trunks are used, the rate will be correspondingly higher. The presence of L/S data will result in gaps in the packet-input data stream. However, this effect will not be considered in order to achieve a worst-case memory requirement.

Speed is not a problem for the input buffers. However, buffer availability could impose a more serious constraint. The buffer must be available for writing every bit time and it must be periodically unloaded into the packet storage area. Two solutions can be immediately proposed:

1) Time multiplex buffer usage - one half bit time for loading, one half for unloading

2) Split the buffer (or double buffer) to perform reading and writing simultaneously

Both solutions are, of course, feasible and involve certain trade-offs. The choice must be made by considering the requirements of the other circuits.

Note that the size of the input buffers (S) is as yet unconstrained. It might range from one bit to one packet in size. As originally conceived each buffer held an entire packet until it had been completely received.

Removal of packet headers can be achieved by a simple demultiplexing (steering) circuit. Headers may be included in the packet storage (PS) if desired.

Because they are relatively infrequent (as little as 3% of a data packet bit stream), header contention for the packet header storage (PH) should not be a problem. Transfer of headers from the input buffers to the packet header storage can be scheduled to avoid conflicts.

The primary problem occurs with the placement of incoming packets into the common packet storage area (PS). With ten bi-directional trunks accessing a single memory, the memory bandwidth must be over 15 megabits per second (10 x 1.544 mbps) under worst case conditions. If bit serial reading and writing were used, a 33ns cycle time would be required ($t_{cyc} = 1/(2 \times 15 \text{ mbps})$). Figure 62 presents the necessary cycle times for PS memories with various word sizes.

Another way of viewing the packet storage memory cycle time requirements is to consider the number of bi-directional trunks that could be supported by a readily available 500ns memory. That information is presented in Figure 63. Naturally, there are other techniques which might be employed to provide additional memory bandwidth, but structuring the data as words (i.e. serial to parallel conversion), is the easiest and most straightforward. This operation could easily be accomplished in the input buffers.

WORD SIZE	PS CYCLE TIME
1 Bit	33 ns
8 Bits	264 ns
10 Bits	528 ns
32 Bits	1056 ns

Figure 62. Required PS Cycle Time for a Node With Ten Bi-Directional Trunks

WORD SIZE	# OF BI-DIRECTIONAL TRUNKS
1 Bit	0.6
8 Bits	5.2
16 Bits	10.4
32 Bits	20.7

Figure 63. Trunk Capacity for a Node Utilizing a PS Memory With a 500 ns Cycle Time

The bandwidth of the packet-storage area must be matched to the total packet bandwidth requirements of the node (including packetized voice). The input and output buffers (S) may tend to alleviate the contention problem as a whole at the packet storage area. The effect will be in direct relation to the ratio of L/S to P/S

within the frame and to the size of the input buffers. However, large input and output buffers are not required for nodes serving a moderate number of trunks (e.g. less than twenty).

PACKETIZED VOICE DATA FLOW AND STORAGE

As voice packets are passed through the network, each node routes them through its packet processing facilities. The voice packets may be buffered in the same storage area as other packet types, but their precedence level should be second only to network control packets. This will minimize delay and therefore skew through the network. Even so, the destination node will require a buffer capable of removing network generated skew. Obviously, the voice samples must be de-compressed (expanded) by some facility at the destination node before they are presented to the user port.

THE ASSOCIATIVE MULTI-ACCESS SWITCH

In conjunction with study task 2 (functional design of integrated switch), several new technologies for the cross-switch path connection were investigated. Among these was a specific switch previously invented at Honeywell and known as the Associative Multi-Access Switch*, or simply AMAS, Figure 64. The objectives of the investigation of this switch were:

- Evaluation of the potential for applying techniques developed as computer technology, specifically associative techniques

- Determination of algorithms for cross-switch path selection and the resulting impact on controller data processing requirements

---

*Multiprocessor Computing Apparatus, U.S.Patent No. 3,521,238, July 21, 1970, D. C. Gunderson.

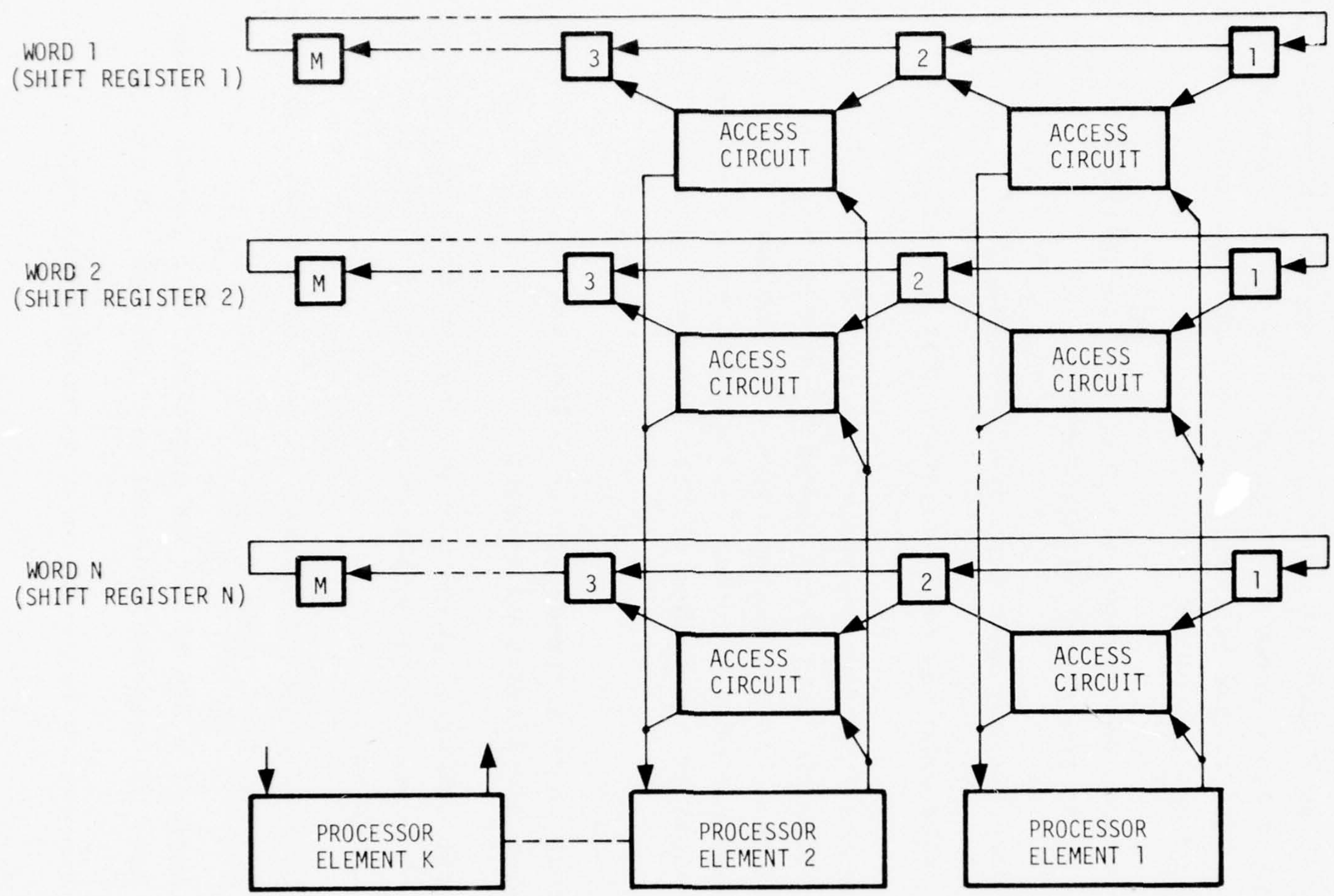


Figure 64. Associative Multi-Access Switch

AMAS is basically a memory array of n words, m bits per word. This is mechanized as n shift registers, m bits each. Processing elements or I/O devices are connected to the shift registers through access circuitry. This access circuitry allows each processing element or I/O device to do the following:

- Perform associative searches (bit serially) on all words (shift registers) simultaneously
- Read out any selected words (bit serially)
- Write into any empty location (bit serially)

In a system configuration as shown in Figure 64, the associative switch provides the path and a temporary storage for messages to be transferred from one processing element to another. The number of processing elements or other devices that can be attached to the switch is equal to the number of bits per word (bits per shift register).

This approach to an interconnection switch was considered to have the following significant advantages:

- The number of paths between any two connected elements is equal to the number of words (shift registers) in the array.
- Loss of a shift register is not catastrophic, but simply reduces the number of paths.
- Loss of a set of access circuitry will result in the loss of only one processing element or I/O device.
- The switch has a cellular structure that makes it

- Expandable
- Suitable for LSI implementation (pin limitations may be a problem)

- The switch is rearrangeable every basic cycle time and, hence, is nonblocking.

Significant areas covered during the investigation of AMAS include:

- Associative addressing techniques
- Self routing data packages
- LSI implementation
- Call setup and takedown
- Call blocking potential

AMAS was determined to be potentially useful in solving several aspects of the cross-switch path problem. A number of problem areas requiring extensive development were identified and studied. Some of the unresolved problems include:

- Speed requirement (faster than trunks)
- Multiple match resolution (empty locations)
- Output contention (active locations)
- Cross switch delay
- Access circuit scope
- Buffer requirements
- Control method

As a result of the AMAS studies many of the AMAS concepts were used as benchmark comparisons during the development of the Associative Call-Buffer switch which has been described.

SECTION 5

SUMMARY OF RESULTS AND RECOMMENDATIONS FOR FUTURE WORK

The most significant results of this study contract include the definitions of:

- A flexible network structure containing homogeneous nodes which can serve as user access points and/or network switching centers
- A network that supports a wide spectrum of user characteristics ranging from the "on-demand, fixed delay" requirements for voice transmission to the "as-available, variable delay" requirements for bulk data transmission
- A line integration technique that allows efficient utilization of transmission bandwidth regardless of network traffic mix

Another key result was the identification and implementation of:

- Nodal functions that are amenable to associative processing techniques

The proposed network structure is not constrained by the hardware in that the system is capable of accommodating links of varying types and speeds ranging from commercial voice grade lines to satellite links. Additionally, the node structure is functionally modular. Only modules that are required for implementation of

the desired nodal function need be included. A user interface is present at a particular node only if needed. There is no hardware penalty for not using a particular feature or type of user interface.

The network is completely transparent to the user. Flexible user-to-network protocols can be provided. Special care has been taken to ensure that bit count integrity of synchronous data is preserved. Precedence level features are included which may be made either visible or invisible to the user. The proposed integration scheme provides new benefits such as the interconnection of incompatible services and the accommodation of different speed data paths.

The approach chosen for voice/data integration on the transmission medium uses variable-size, fixed-position time slots within a constant interval framework. An absolute time reference allows bit count integrity to be maintained and synchronous data to be transmitted without the time slippage problems which can occur with packet-only integration techniques. Variable-size slots permit efficient bandwidth utilization by allowing the slot size to be closely matched to the user requirement. The "clean fill" of packet data between line switched data slots and across frames results in further bandwidth utilization. Compressible voice is handled with an attendant savings in bandwidth (packetized voice).

A switch implementation for line switched data has been designed which uses associative processing techniques. The switch is simple, mudular, and expandable. It can handle variable-speed trunks and exhibits a fixed, minimal delay. Its decentralized control is both simple and reliable. Further study is required to provide hardware resource sharing and combined line switched data and packet switched data processing within the node.

There are several areas that require further work. Common hardware modules and nodal processing techniques should be developed which will result in total integration within the node. The applicability of associative and parallel processing techniques to the local access functions should be investigated further. For practical considerations, a plan is required which describes an evolutionary transition from planned systems such as the TTC-39 switch to totally integrated, digital switching and on to future systems including satellite switching centers. The reliability aspect of a proposed implementation must also be studied further.

REFERENCES

1. Davies, D.W. and D.L.A. Barber, Communication Networks for Computers, John Wiley and Sons, New York, N.Y., 1973.

2. Draft Specification (Type A) for AUTODIN II Phase I, February 1975.

3. Draft 3: American National Standards Institute (ANSI), Advanced Data Communication Control Procedures (ADCCP) -- Independent Numbering, X3S34/589, March 27, 1975.

4. DCS AUTODIN Interface and Control Criteria, DCAC 370-D175-1.

5. DoD Data Internet Study, Phase II Report.

6. P. Zafiropulo, "Flexible Multiplexing for Networks Supporting Line-Switched and Packet-Switched Data Traffic," ICCC 1974 Conference, Stockholm, Sweden.

APPENDIX

CONTROL PACKET DESCRIPTIONS

(1) Type 1 Control Packet: L/S Slot Reservation Command/Request

Content:

Field	Size	
• Header	64	bits
• Control Packet Type	4	
• Called Party Number (12 digits)	48	
• Station Characteristics	48	
• Class I Precedence Level	3	
• Class IA Call Number	14	
• Slot Specifications	100	
	281	bits
	36	bytes

Description:
The "L/S slot reservation command/request" control packet is used for line switched (Class IA) call setup. It communicates destination and slot reservation information from the line master node to the line slave node.

The 64-bit Header is previously defined in Figure 12. The header contains a 4-bit packet type field which indicates that this is a control packet. The 4-bit control packet type field shown above indicates that this is a Type 1 control packet. The 48-bit called party number is used by the network nodes to determine the appropriate call path to the called party. The station characteristics

include data type, security mode, voice quality, etc . The Class I precedence level ranges from 1 through 5. The Class IA call number consists of an 8-bit calling port number assigned by the node and a 6-bit calling node address. The Class IA call number is uniquely identified with the call being set up in the network. The 100-bit slot specifications field has five 20-bit subfields. Each 20-bit subfield consists of an 11-bit number for "slot starting byte address" and a 9-bit number for "slot size in bytes". It is assumed that any Class IA call cannot occupy more than five slots. (If the traffic on a trunk is such that a Class IA call has to be divided into more than five slots, then that call will not be set up.) The 11-bit number for slot starting byte address is capable of addressing 2048 bytes. The 9-bit number for slot size can specify up to 512 bytes. For 10-msec frames (15,440 bits each) on T1 lines and assuming that the highest speed Class IA call is 300 kbps (or 3000 bits per 10-ms frame), the 11-bit number for slot starting byte address and the 9-bit number for slot size in bytes are sufficient.

The total length of a Type 1 control packet is 281 bits. In order to put every packet into a multiple of bytes, a Type 1 control packet occupies 36 bytes.

(2) Type 1a Control Packet: L/S SLot Reservation Command

Content:

Field	Size
• Header	64 bits
• Control Packet Type	4

• Class IA Call Number	14
• Slot Specifications	100
Total	182 bits (23 bytes)

Description:

The "L/S slot reservation command" control packet is the normal, reverse direction response to a type 1 control packet. It communicates slot reservation information toward the calling node.

Note that the Class IA call number is uniquely associated with the call being set up; the node that receives a Type 1a control packet can properly process it for the call being set up.

(3) Type 2 Control Packet: L/S SLot Reservation Request Denied

Content:

Field	Size
• Header	64 bits
• Control Packet Type	4
• Class IA Call Number	14
Total	82 bits (11 bytes)

Description:

The "L/S slot reservation request denied" control packet is sent toward the calling node when time slots are not available in that direction.

(4) Type 3 Control Packet: L/S Unreserve Command

Content:

	Field	Size
•	Header	64 bits
•	Control Packet Type	4
•	Class IA Call Number	14
	Total	82 bits
		(11 bytes)

Description:

The "L/S slot unreserve command" control packet indicates that no forward routes are available for the call being set up and that alternate route selection is required at a previous node.

(5) Type 4 Control Packet: L/S Slot De-Reservation Command

Content:

	Field	Size
•	Header	64 bits
•	Control Packet Type	4
•	Class IA Call Number	14
	Total	82 bits
		(11 bytes)

Description:

The "L/S slot de-reservation command" control packet is used for line switched call takedown. It is also used for line preemption.

(6) Type 5 Control Packet: P/S Transmission Connection Request

Content:

	Field	Size
•	Header	64 bits
•	Control Packet Type	4
•	Called Party Number (12 digits)	48
•	Station Characteristics	48
	Total	164 bits (21 bytes)

Description:

The "P/S transmission connection request" control packet is sent by a source (calling) node to a destination (called) node to check the "on/off" status of the called port.

The called party number is used by the destination (called) node to identify the called party. The station characteristics include data type, security mode, message security class, etc. The abbreviated source port number in the header contains the "calling port number" assigned by the calling node.

(7) Type 6 Control Packet: P/S Transmission Connection Answer

Content:

	Field	Size
•	Header	64 bits
•	Control Packet Type	4
•	Called Party Status (on/off)	1
	Total	69 bits (9 bytes)

Description:

The "P/S transmission connection answer" control packet is sent by a destination (called) node of an attempted P/S one-direction logical connection to the source (calling) node (as a response to a Type 5 Control Packet) to indicate the on/off status of the called port.

The abbreviated source port number in the header contains the "calling port number" assigned by the calling node. That is, it contains the same number as in the corresponding field of the Type 5 control packet to which this Type 6 control packet is responding.

(8) Type 7 Control Packet: Multi-Packet Message Transmission Request

Content:

Field	Size
• Header	64 bits
• Control Packet Type	4
Total	68 bits (9 bytes)

Description:

The "multi-packet message transmission request" control packet is sent by a source (calling) node to a destination (called) node to request allocation of buffer space of one multi-packet message at the destination (called) node.

A similar request is not needed for transmission of a single-packet message.

(9) Type 8 Control Packet: Ready to Receive Multi-Packet Message

Content:

Field	Size
• Header	64 bits
• Control Packet Type	4
Total	68 bits (9 bytes)

Description:

The "ready to receive multi-packet message" control packet is sent by a destination (called) node to a source (calling) node in response to a Type 7 control packet, or after the destination node receives, reassembles and outputs a complete multi-packet message. This control packet allows the source (calling) node to start sending a multi-packet message. When sent in response to the receipt of a complete multi-packet message, the message number field in the header contains the message number for the multi-packet message just received.

10) Type 9 Control Packet: Confirm Acceptance of a Single-Packet Message

Content:

Field	Size
• Header	64 bits
• Control Packet Type	4
Total	68 bits (9 bytes)

Description:

The "confirm acceptance of a single-packet message" control packet is sent by a destination (called) node to a source (calling) node to inform the source node that a single-packet message, whose message number is in the message number field of the header, has been accepted by the destination (called) node.

(11) Type 10 Control Packet: Packetized Voice (P/V) Call Setup Request/Command

Content:

Field	Size
• Header	64 bits
• Control Packet Type	4
• Called Party Number (12 digits)	48
• Station Characteristics	48
• Class I Precedence Level	3
• Class IB Call Number	14
Total	181 bits (23 bytes)

Description:

The "P/V call setup request/command" control packet is used for setting up a call path segment between two adjacent nodes for a P/V call (Class IB).

The fields in the Type 10 control packet which are the same as in the Type 1 control packet (L/S slot reservation command/request) have the same definitions. The Class IB call number, similar to Class IA call number in

Type 1 control packet, consists of an 8-bit calling port number assigned by the calling node and a 6-bit calling node address. Because L/S calls and P/V calls are handled as two completely separate categories in the network, the calling port numbers for a L/S call and a P/V call are assigned independently (i.e., a node may establish up to 256 local L/S calls and up to 256 local P/V calls at any given time.)

(12) Type 11 Control Packet: P/V Call Setup Request Denied

Content:

	Field	Size
•	Header	64 bits
•	Control Packet Type	4
•	Class IB Call Number	14
	Total	82 bits (11 bytes)

Description:

The "P/V call setup request denied" control packet is generated and sent to the immediately preceeding node (on a partially set up path of a P/V call) to convey that the attempted P/V call (identified by the Class IB call number) is blocked at the current node.

(13) Type 12 Control Packet: P/V Call Takedown Command

Content:

	Field	Size
•	Header	64 bits
•	Control Packet Type	4
•	Class IB Call Number	14
	Total	82 bits (11 bytes)

Description:

The "P/V call takedown command" control packet is used for taking down a call path segment between two adjacent nodes during the P/V call takedown process.